VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ
FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY
INSTITUTE OF INFORMATICS

NÁVRH ZABEZPEČENÍ PRŮMYSLOVÉHO ŘÍDÍCÍHO SYSTÉMU
INDUSTRIAL CONTROL SYSTEM SECURITY DESIGN

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — **Bc. Matěj Strnad**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. Petr Sedlák**
SUPERVISOR

BRNO 2019

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA PODNIKATELSKÁ

Zadání diplomové práce

Ústav:	Ústav informatiky
Student:	**Bc. Matěj Strnad**
Studijní program:	Systémové inženýrství a informatika
Studijní obor:	Informační management
Vedoucí práce:	**Ing. Petr Sedlák**
Akademický rok:	2018/19

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává diplomovou práci s názvem:

Návrh zabezpečení průmyslového řídícího systému

Charakteristika problematiky úkolu:

Úvod
Vymezení problému a cíle práce
Analýza současného stavu
Teoretická východiska práce
Vlastní návrh řešení
Závěr
Seznam použité literatury
Přílohy

Cíle, kterých má být dosaženo:

Cílem práce je navrhnout řešení zabezpečení průmyslového řídícího systému. Cílovým výstupem je návrh řešení, které splňuje požadavky investora.

Základní literární prameny:

COLBERT, Edward J. Cyber-security of SCADA and other industrial control systems. New York, NY: Springer Science+Business Media, 2016. ISBN 978-33-1932-123-3.

KOLOUCH, Jan a Pavel BAŠTA. CyberSecurity. Praha: CZ.NIC, z.s.p.o., 2019. ISBN 978-80-881-8-31-7.

ONDRÁK, Viktor, Petr SEDLÁK a Vladimír MAZÁLEK. Problematika ISMS v manažerské informatice. Brno: Akademické nakladatelství CERM, 2013. ISBN 978-80-7204-872-4.

TANENBAUM, Andrew a David WETHERALL. Computer networks. 5. vyd. Boston: Pearson Prentice Hall, 2011. ISBN 0-13-212695-8.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2018/19

V Brně dne 28.2.2019

L. S.

doc. RNDr. Bedřich Půža, CSc.
ředitel

doc. Ing. et Ing. Stanislav Škapa, Ph.D.
děkan

Abstrakt

Náplní diplomové práce je návrh bezpečnostních opatření pro zabezpečení průmyslového řídícího systému. Obsahuje analýzu komunikačního prostředí a specifik průmyslových komunikačních systémů, srovnání dostupných technologických prostředků a návrh řešení v souladu s požadavky investora.

Abstract

The subject of the master's thesis is a design of security measures for securing of an industrial control system. It includes an analysis of characteristics of communication environment and specifics of industrial communication systems, a comparison of available technological means and a design of a solution according to investor's requirements.

Klíčové slova

intrusion detection system, intrusion prevention system, průmyslové řídící systémy, bezpečnostní monitoring

Key words

intrusion detection system, intrusion prevention system, industrial control system, security monitoring

Bibliografická citace

STRNAD, Matěj. *Návrh zabezpečení průmyslového řídícího systému* [online]. Brno, 2019 [cit. 2019-05-08]. Dostupné z: https://www.vutbr.cz/studenti/zav-prace/detail/119834. Diplomová práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Petr Sedlák.

Čestné prohlášení

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 10. května 2019

……

podpis studenta

Poděkování

Rád bych poděkoval svému vedoucímu Ing. Petrovi Sedlákovi za trpělivost, kritiku, cenné informace, rady a připomínky, které umožnily vznik této práce.

OBSAH

ÚVOD

V průběhu posledních desetiletí jsme mohli zaznamenat prudký rozvoj informačních a komunikačních technologií (ICT) a jejich integraci do každé oblasti života člověka – vědních oborů, průmyslu, dopravy, školství, energetiky i socioekonomické sféry. Díky těmto změnám došlo k rozvoji a zvýšení celkové efektivity ve všech oblastech lidského bádání a ICT je dnes nedílnou součástí základní infrastruktury moderního státu.

Integrace informačních a komunikačních technologií do státní infrastruktury nese určitá rizika. Technologický pokrok šel kupředu a rizika a dopady této integrace na jednotlivce i společnost se začaly brát v potaz až dodatečně. Trochu opožděně proti samotnému rozvoji ICT se rozvíjí i obor informační a kybernetické bezpečnosti, jehož cílem je zajistit bezpečnost informací i procesů v naší společnosti, v podnikovém prostředí i v životě jednotlivců.

Různá oborová prostředí vyžadují specifická opatření a technologické postupy. V oblasti průmyslové výroby je kontinuita (dostupnost) výrobního procesu na prvním místě a bezpečnostní opatření nesmí být s tímto faktem v rozporu. Tato práce se zabývá právě návrhem bezpečnostních opatření v prostředí průmyslových řídících systémů.

VYMEZENÍ PROBLÉMU A CÍLE PRÁCE

Cílem této práce je navrhnout opatření pro zvýšení celkové bezpečnosti z hlediska informačních a komunikačních technologií. Návrh cílí na řídící systémy využívané k řízení energetické distribuční soustavy, které dle zákonem definovaných kritérií spadají do kritické informační infrastruktury státu – vztahuje se na ně tedy Zákon č. 181/2014 Sb., o kybernetické bezpečnosti (novelizovaný č. 205/2017 Sb.).

Výstup této práce se skládá ze dvou částí – analytické a návrhové. Analytická část obsahuje rozbor požadavků investora, vlastností řídícího systému, jeho komunikace a dostupných technologických prostředků. Návrhová část představuje řešení technologických i organizačních opatření. Teoretický rozbor objasňuje specifické pojmy používané v analytické a návrhové části.

1 ANALÝZA SOUČASNÉHO STAVU

Současný stav je v této kapitole chápán v širším pojetí a zahrnuje investorovy požadavky, popis prostředí průmyslového řídícího systému v distribučních rozvodnách, zhodnocení dostupných technologií z hlediska využití v navrhovaném řešení a výchozí předpoklady, na nichž je práce založena.

1.1 Analýza požadavků investora

S investorem bylo konzultováno zadání práce a sepsány jeho požadavky. Investor požaduje:

- Návrh uceleného řešení pro zabezpečení průmyslového řídícího systému, které může být kombinací jak technologických, tak organizačních opatření.
- Ověření využitelnosti open source nástrojů pro detekci nežádoucí aktivity, případně je zakomponovat v návrhu řešení.
- Řešení musí být navrženo s ohledem na specifika průmyslového prostředí s vysokým požadavkem na jeho dostupnost a s ohledem na specifické komunikační protokoly, které jsou v rámci řídících systémů distribučních rozvoden využívány.
- Řešení nemusí obsahovat návrh opatření pro veškeré hrozby působící na průmyslový řídící systém, cílem je dodat především principiální řešení.
- Preferovaný operační systém pro běh bezpečnostních technologií je OS Linux.

Pro potřeby zpracování návrhu byla dodána řádná dokumentace systému a reprezentativní záznam síťového provozu, který byl pořízen v reálném prostředí. Pro dané prostředí byla zpracována analýza rizik. V rámci navrhovaných opatření na zjištěná rizika je požadováno nasazení technologií pro monitoring událostí a síťových toků v rámci lokální sítě ICS. Návrh technologií pro takový monitoring by měl být zahrnut v této práci.

1.2 Analýza průmyslového prostředí

Tato kapitola obsahuje popis průmyslového řídícího systému (dále jen ICS) v rámci sítě LAN, na kterou budou aplikována bezpečnostní opatření. Kapitola vychází z dodané dokumentace, osobních konzultací a analýzy záznamu síťového provozu.

Průmyslové sítě se do značné míry liší od klasických komerčních sítí a návrh bezpečnostních opatření vyžaduje odlišný přístup. Po konzultaci s investorem vyplývají pro síť ICS následující specifika:

- Komponenty ICS mají vyšší odolnost, rozsah provozní teploty, způsob napájení a liší se v dalších parametrech, které jsou podřízené primárně větší dostupnosti, nižší poruchovosti. Kvůli tomu mají i vyšší pořizovací cenu, obvykle mají garantovanou funkčnost jen v rámci produktového portfolia jednoho dodavatele, a proto je velmi obtížné takové zařízení nahradit (např. za zařízení, která lépe splňuje kritéria kybernetické bezpečnosti).
- V prostředí ICS je vyšší požadavek na dostupnost řízeného procesu, což značně znesnadňuje vulnerability management a patch management – povýšení verze firmware/operačního systému/aplikace musí probíhat ve spolupráci správců několika různých aktiv, vyžaduje mnohem náročnější testování po uvedení do provozu.
- U některých průmyslových aplikací výrobce garantuje funkčnost jen pro danou verzi a build operačního systému a v takovém případě vulnerability a patch management téměř znemožňuje.
- Očekávaná životnost systémových komponent se pohybuje v rozmezí 8-20 let oproti standardním 5 rokům v komerčním prostředí, ale zranitelnosti pro systémové komponenty mohou být zjištěny v mnohem kratších časových intervalech – pokud tedy nelze okamžitě povýšit verzi komponenty, je potřeba monitorovat pokusy o zneužití takových zranitelností a dané slabiny aktiv pečlivě evidovat.
- Kvůli vysokému požadavku na dostupnost systému není z hlediska správců žádoucí nasazení IPS/Firewallů a podobných prvků, které:
    - přidají do komunikačního kanálu další prvek, který může selhat,

- kvůli nesprávné konfiguraci ohrozí funkcionalitu celkového řídícího systému.

Preferované řešení je tedy pasivní monitoring sítě prostřednictvím mirror/SPAN portů a nasazení IDS.

- Síťový provoz v ICS vyžaduje nižší šířku pásma – menší objemy přenášených dat, protokoly často přenášejí data v binárním formátu.
- Kvůli potřebě přiblížení se real-time řízení v lokální síti je přenášeno především velké množství krátkých rámců, které obsahující telemetrická data vztahující se k jednotlivým monitorovaným datovým bodům.

Návrh řešení bezpečnosti musí být podřízen těmto vlastnostem.

1.2.1 Popis síťových uzlů

V rámci ICS se může vyskytovat několik ochranných relé – obecně **IED** (*Intelligent Electronic Devices*), programovatelných řídících jednotek (**PLC** – *Programmable Logic Controller*), koncentrátorů / sběračů signálů / **RTU** (*Remote Terminal Unit*) a řídící terminály **HMI** (*Human Machine Interface*). Lokální síť dále zpravidla obsahuje průmyslové prvky komunikační infrastruktury – přístupové L2 a L3 switche, případně dodatečnou gateway zajišťující směrování komunikace mezi lokálními koncovými uzly a centrálním SCADA systémem. Pro eliminování časové nejistoty v lokální síti jsou někdy přítomny i časové servery.

V rámci sítí s vysokou dostupnosti jsou běžné síťové topologie obsahující redundantní linky. Ty jsou řízeny buď pomocí RSTP (*Rapid Spanning Tree Protocol*) nebo vhodnějšími protokoly pro řízení redundance v závislosti na konkrétní fyzické topologii – příkladem může být HSR (*High Availability – Seamless Redundancy*), PRP (*Parallel Redundancy Protocol*) nebo MRP (*Media Redundancy Protocol*). Redundantně mohou být zapojeny jak prvky komunikační infrastruktury, tak koncové uzly. Lokální síť může být připojena do sítě WAN buď jedním zařízením, případně může být připojena redundantně.

1.2.2 Rozbor komunikace

Poskytnutý záznam síťového provozu byl pořízen v průběhu 7 dní a výsledný soubor dosahuje velikosti 11.49 GB. Ze záznamu bylo potřeba zjistit, jaké protokoly obecně mohou být v prostředí ICS využívány a pomocí jakých protokolů komunikují jednotlivá zařízení mezi sebou nebo mimo lokální síť. Tato zjištění poslouží pro určení vhodných pravidel pro budoucí IDS (*Intrusion Detection System*).

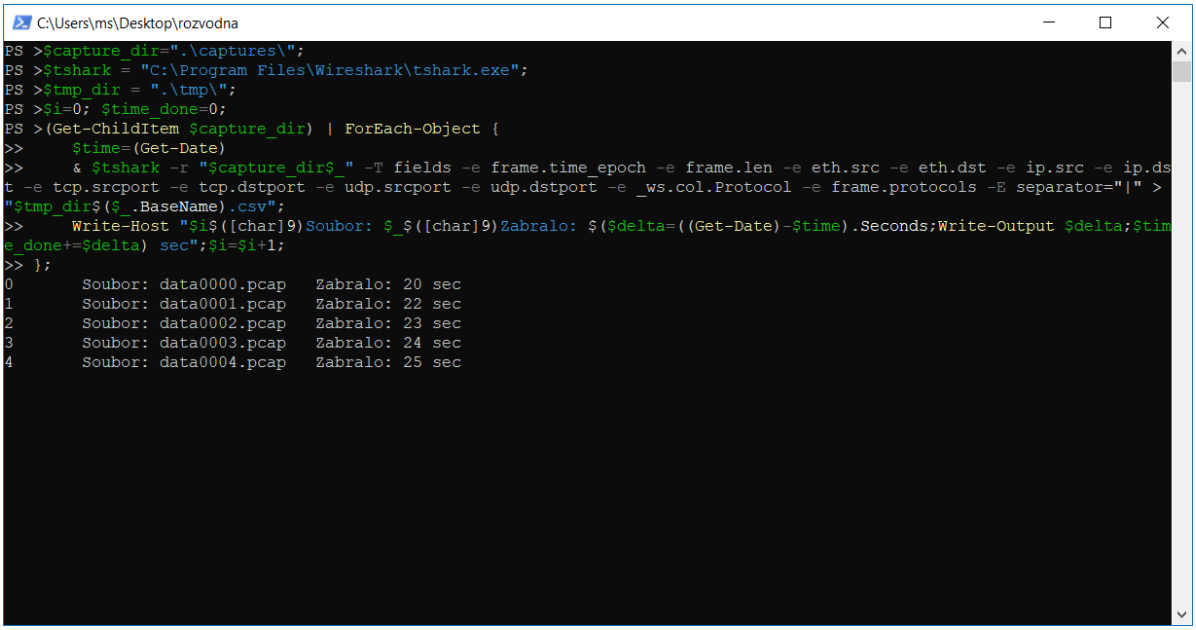
Záznam byl pořízen prostřednictvím utility *tcpdump* a z praktických důvodů byl rozdělen na kolekci menších souborů (o velikosti 50 MB) ve formátu PCAP. Problémem větších záznamů síťového provozu je jejich následné vyhodnocení. Program *Wireshark* a přibalené utility jsou pro takovou analýzu vhodným nástrojem. Program obsahuje vše nutné pro práci se záznamy síťového provozu včetně statistických nástrojů a pokročilých filtrů (19).

Otevření celého záznamu v programu je problematické, protože jak *Wireshark*, tak jeho příkazová utilita *tshark* při otevření celý záznam nahrají do RAM, a s jeho rostoucí velikostí je jakákoliv operace v programu progresivně výpočetně náročnější. Příliš velké záznamy povedou jedině k pádu programu. Analyzovat několik vybraných souborů z kolekce by nebylo dostatečně reprezentativní a postrádalo smysl.

Proto bylo přistoupeno ke skriptovému zpracování. Napsat program vhodný pro takovou analýzu by bylo časově příliš náročné a nad rámec této práce. Aby bylo dosaženo výsledků relativně rychle a efektivně, byly použity vyšší skriptovací jazyky (Python, Powershell) v kombinaci s utilitou tshark (20, 21).

Vzorek byl uložený v binárním formátu PCAP, pro který má python k dispozici knihovny, jako například Scapy nebo PyShark. Pokus o využití obou dvou ztroskotalo na příliš pomalém zpracování. Tshark má možnost exportovat jednotlivá pole a informace pro každý rámec do plochého textového CSV souboru, nicméně neobsahuje vhodné parametry pro toto zpracování, proto byla využit v kombinaci s Pythonem, který jednoduchý textový soubor zpracuje rychleji než data v binární podobě (22, 23).

Jako nejrychlejší způsob zpracování se ukázalo sériové volání utility tshark pro export potřebných polí pomocí PowerShellu a následné zpracování vygenerovaných textových souborů pomocí pythonového skriptu. Ten vyexportoval seznam unikátních komunikací v rámci vzorku, seznam použitých protokolů, maximální počet přenesených rámců a bitů v jedné vteřině a dodatečně i seznam všech unikátních IP a MAC adres (pro určení, jestli se v rámci lokální sítě nenachází nezdokumentované zařízení). Ukázka volání tsharku pomocí PowerShellu viz příloha 2.



Obrázek 1: Export polí ze vzorku síťového provozu pomocí tsharku a Powershellu. (Zdroj: Vlastní tvorba)

Pro účely zpracování velkých síťových vzorků lze alternativně zapojit programy Zeek NSM nebo NetworkMiner, ale i využití těchto programů má své úskalí (např. Zeek není schopen identifikovat tolik protokolů jako Wireshark).

V prvé řadě bylo nutné určit, jaké komunikační protokoly jsou v rámci lokální sítě používány. Přehled protokolů, počet zaznamenaných rámců a jejich procentuální podíl na celém vzorku – příklad viz tabulka 1. V tabulce jsou využity názvy, které protokolům přidělí Wireshark, příkladem může být 104apci a 104asdu označující v obou případech protokol dle normy IEC 60870-5-104.

Tabulka 1: Přehled zaznamenaných komunikačních protokolů. (Zdroj: Vlastní tvorba)

Protokol	**Počet rámců**	**%**	**Poznámka**
GOOSE	41822979	48,56	protokol dle normy IEC 61850
TCP	35487370	41,20	
ARP	2306083	2,68	
104asdu	1870416	2,17	protokol dle normy IEC 60870-5-104
STP	1764895	2,05	
104apci	1330701	1,54	protokol dle normy IEC 60870-5-104
SUMA	86130252	100	
Velikost souboru	11.49 GB		
Average bitrate	0.235 Mbps		
Maximum bitrate	0.725 Mbps		

Zároveň s protokoly byly identifikovány i jednotlivé špičky v provozu, určen maximální objem přenášených dat 0.725 Mbps a průměrný počet paketů za sekundu 235 pps – jedná se o klidovou komunikaci, kdy z lokální sítě probíhá primárně odesílání telemetrie. Maximální propustnost síťového stacku linuxového jádra je dle některých zdrojů odhadovaná na přibližně 480 000 pps, je zde tedy ponechána více než dostačující kapacitní rezerva pro budoucí monitorovací server (24).

Dalším z výstupů analýzy byl seznam jednotlivých unikátních komunikací mezi dvěma uzly pomocí jednotlivých protokolů. Ukázka tohoto výstupu viz tabulka 2. Tento výstup poslouží pro sestavení pravidel IDS a dalších monitorovacích aplikací – např. pro danou IP adresu bude v konfiguraci / politice povolena komunikace jen pro vybraný seznam protokolů a jakákoliv jiná komunikace bude vyhodnocena jako anomální a potenciálně nežádoucí.

Tabulka 2: Ukázka přehledu komunikací zachycených ve vzorku síťového provozu. (Zdroj: Vlastní tvorba)

Zdroj	Cíl	Protokol	Počet rámců
172.16.20.20	172.16.10.101	TLSv1	45
172.16.20.20	172.16.20.20	TLSv1	78
172.16.10.101	172.16.10.15	MMS	331
172.16.10.101	172.16.10.101	MMS	692
172.16.10.15	172.16.10.16	MMS	218
172.16.10.101	172.16.10.101	MMS	502
172.16.10.16	172.16.10.101	MMS	4
172.16.10.82	172.16.10.252	SSHv2	47
172.16.20.20	172.16.20.20	SSHv2	119
172.16.10.252	172.16.20.20	SSHv2	222
172.16.10.253	172.16.10.253	SSHv2	2
172.16.20.20	172.16.10.101	TLSv1	45

Dalšími výstupy byly seznamy unikátních IP a MAC adres, seznam souborů s naměřenými špičkami a seznam protokolů za jednotlivé soubory (pro usnadnění navigace mezi jednotlivými soubory záznamu). Tyto výstupy mají praktický význam pro hlubší analýzu na straně příjemce tohoto návrhu a v rámci této práce nemá smysl je podrobněji rozebírat.

1.3 Prostředky pro monitoring událostí

Existuje velké množství komerčních řešení pro bezpečnostní monitoring jak na úrovni operačního systému (*Host-based Intrusion Detection Systém – HIDS*), tak na úrovni síťové komunikace (*Network Intrusion Detection Systém – NIDS*). Stejně tak existuje řada open source IDS / IPS nástrojů, které jsou distribuovány zdarma pod svobodnými licencemi (například BSD nebo GPL). V rámci práce budou využity síťově orientované IDS.

U NIDS je jedním z důležitých parametrů podpora parsování protokolu, nad kterým chceme provádět kontrolu. Nepřítomnost parseru ještě neznamená, že pro daný protokol nelze

definovat v NIDS pravidla. Parser NIDS pouze umožňuje dekódování protokolu v reálném čase a díky němu lze psát snadněji pravidla. Většina NIDS podporuje běžné komerční protokoly, u průmyslových protokolů je podpora většinou pouze pro Modbus, Ethernet/IP a DNP3.

Zásadním problémem většiny dnes dostupných řešení je jejich orientace na klasické komerční IT. Existuje však řada výrobců, která dodává specializované IDS systémy pro monitoring průmyslových řídících systémů. Příkladem komerčních výrobců může být CyberX, Forescout nebo Fortinet. Někteří výrobci používají pro vývoj svého řešení open source IDS. Jejich zařízení bývají ale podstatně dražší a jejich řádné srovnání by vyžadovalo nasazení do zabezpečované sítě a řádné testování vybraných scénářů útoků, funkcionalit a ladění jejich nastavení, aby na závěr bylo možno objektivně vyhodnotit, které řešení je vhodné pořídit.

Prodejci těchto IDS slibují implementované celé řady protokolů, nicméně podrobnosti této implementace nemusí být úplně známy, dokud se na ně zákazník přímo nezeptá. Podstatnou otázkou, kterou by mělo testování komerčních IDS zodpovědět, je, jestli anomálii nebo specifický útok IDS skutečně zachytí i v případě, že k němu dojde u zařízení, které je na whitelistu. *Tato práce bude zaměřena na určení využitelnosti open source IDS pro monitoring průmyslových komunikací.*

1.3.1 Snort

Snort je síťově orientovaný IDS / IPS systém schopný provádět analýzu a záznam síťového provozu v reálném čase. V současné době je ve vlastnictví společnosti Cisco. Je vyvíjen od roku 1998. Analýzu provádí na základě definovaných pravidel (signatur). Tato pravidla mohou být manuálně definovaná uživatelem, stažena z veřejných knihoven anebo lze zakoupit prémiovou sadu pravidel společně s komerční podporou. V rámci prémiové sady pravidel od Cisco Talos jsou i pravidla pro protokoly IEC 60870-5-104 a MMS, které jsou využívány v komunikaci zabezpečovaného prostředí (25).

V současné době jsou vyvíjeny dvě verze – Snort 2.x a Snort 3.0, kdy novější verze se snaží vyřešit nedostatky té starší. Protože vývoj Snortu začal před dvaceti lety, ve verzi 2.x je aplikace schopná fungovat pouze v jednovláknovém režimu, což komplikuje nasazení ve vysokorychlostních sítích (26).

Výhodou a zároveň nevýhodou je jednoduchost pravidel pro detekci. Syntaxe a struktura pravidel mohla být v době počátků vývoje tohoto IDS dostačující, ale s rostoucí složitostí útoků v dnešním ICT je buď složité nebo nemožné některé typy útoků detekovat pomocí standardních pravidel (26).

Protože jsou pravidla Snort jednoduchá, bývají často v přílohách informativních oběžníků a článků upozorňujících na nové hrozby. Snort je z open source NIDS nejpopulárnější a má širokou uživatelskou komunitu. Značná výhoda je i jeho snadná instalace a správa (26).

Snort 3.x usiluje o vyřešení některých nedostatků, kterými trpí starší verze. Kromě mnoha nových funkcí byla přidána podpora multivláknového režimu a integrovaný skriptovací Lua engine, který umožňuje rozšířit základní pravidla o detekci složitějších scénářů. V reakci na nedostatky verze dva se již dříve začal vyvíjet další open source NIDS – Suricata. Otázkou tedy zůstává, jaký smysl má v budoucnu existence dvou funkčně srovnatelných open source IDS. Protože Snort 3 je v době psaní této práce ještě v beta verzi, není vhodný pro produkční nasazení (25).

1.3.2 Suricata

Suricata je síťově orientovaný open source IDS / IPS systém. Podporuje stejný formát pravidel jako Snort a sady pravidel pro Snort i Surikatu jsou z větší části kompatibilní (drobné výjimky viz. zdroj 27). Jejím hlavním zdrojem signatur je knihovna Emerging Threats (více viz zdroj 28). Od roku 2009 je vyvíjena OISF (*Open Information Security Foundation*) (29).

Oproti Snortu je vyvíjena jako multivláknová aplikace, obsahuje skriptovací engine Lua pro detekci složitějších útoků, logování v JSON formátu (výhoda pro snazší zpracování dalšími

analytickými nástroji). Také má automatickou detekci některých aplikačních protokolů (např. HTTP, SMB, DNS, FTP, …) (29).

Oproti Snortu má menší uživatelskou základnu, protože ale podporuje více vláken, obsahuje Lua scripting engine, je kompatibilní se Snort pravidly a je s těmito funkcemi distribuována ve stabilní verzi, jeví se jako vhodnější kandidát na nasazení než Snort. Dle některých zdrojů dokáže zpracovat přibližně 1 Gbps na jednu instanci (30).

1.3.3 Zeek

Zeek Network Security Monitor není pouze NIDS zaměřený na detekci záškodnické aktivity v síti, ale je to spíše flexibilní platforma / framework umožňující dlouhodobou hlubší analýzu síťového provozu. Je postupně vyvíjen už od roku 1995, přestože většinu času pod jménem Bro a až v nedávné době byl přejmenován na Zeek (31).

Zeek podporuje detekci pomocí signatur (nekompatibilní si Snortem a Suricatou), ale není to jeho primární zaměření. Primárně monitoruje navázaná spojení v síti a detekuje anomálie v síťovém provozu. K definování monitoringu je využíván vlastní skriptovací jazyk, pomocí kterého lze pro jednotlivé události nadefinovat patřičné akce. (26).

Je velmi flexibilní v tom, co zaznamenává – nemusí se jednat pouze o zasílání varovných zpráv o detekované anomálii do centrálního úložiště. Zeek je velmi dobrý v dlouhodobém zaznamenávání metadat o navázaných spojeních, která ukládá v snadněji zpracovatelném formátu, než by byl plný záznam síťového provozu. Dovede uchovávat metadata jak o obecných spojeních nebo anomáliích, tak i podrobnosti k některým aplikačním protokolům – např. k SSH protokolu v čisté instalaci eviduje typ a verzi serveru a klienta, verzi protokolu, typ algoritmu pro výměnu klíčů, aj. Tato evidence metadat by v případě řešení budoucích incidentů CERT/CSIRT týmu mohla usnadnit práci (26, 31).

Zeek je velmi snadno rozšiřitelný (na rozdíl od Snorta nebo Suricaty) – pro jakoukoliv funkcionalitu lze napsat skriptovou politiku, případně nahrát již napsaný modul. Jeho

úskalím je podstatně menší množství dostupných materiálů a scriptů. Zatímco Snort i Suricata mají k dispozici veřejné knihovny pravidel, hlavním zdrojem scriptů pro Zeek je GitHub společně s těmi obsaženými ve standardní instalaci. Společnost Corelight se snaží vyřešit tento problém tak, že dodává řešení s placenou podporou a snazším nasazením (26, 32).

Hloubková analýza síťového provozu má u Zeeku za následek větší nároky na výpočetní výkon. Čím náročnější detekce je požadována, tím větší dopad na výpočetní výkon. Zeek je jednovláknová aplikace, má ale nativní podporu pro clustering včetně jeho hromadné správy. Lze ho tedy nasadit i ve vysokorychlostních sítích. Některé zdroje uvádějí detekci pro přibližně 250 Mbps na jednu instanci Zeeku (25, 30).

Navzdory celkové flexibilitě, kterou Zeek CERT/CSIRT týmu poskytuje, vyžaduje hodně přizpůsobování a úprav, aby bylo dosaženo optimálního monitoringu. Klade velké nároky na správce, kteří kromě řady konfigurací potřebují znát i specifický programovací jazyk (26).

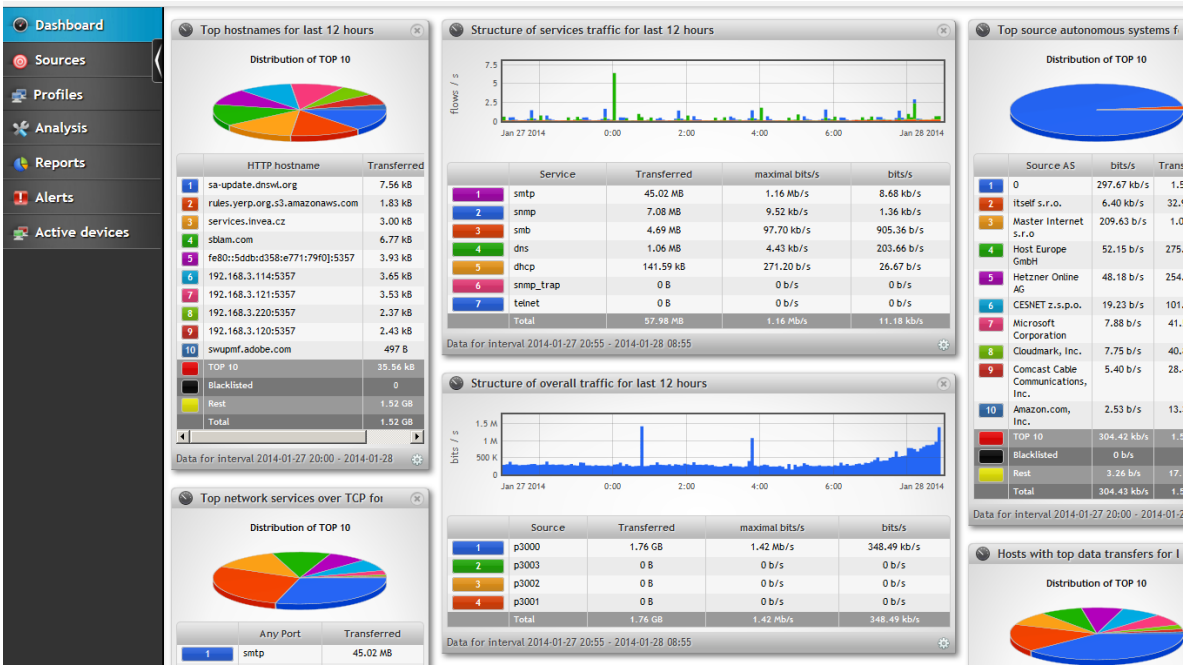
Díky své orientaci na anomálie a obecný monitoring umožňuje Zeek pokrýt jednak sofistikovanější scénáře útoku a zároveň detekovat nové typy škodlivé aktivity v síti. Signatury jsou psané a distribuované pro známé hrozby a až 20 % incidentů v infrastruktuře může být zapříčiněných novým typem útoků (26).

1.3.4 Prostředky pro monitoring síťových toků

Bezpečnostní monitoring lze opřít o zaznamenané události a také o informace o síťových tocích. Důvodem pro monitoring síťových toků je fakt, že některé druhy kompromitace infrastruktury se mohou projevit pouze anomálií v síťové komunikaci, kterou IDS nemusí detekovat (na rozdíl od systémů, které např. statisticky vyhodnocují síťové toky). Tyto dva zdroje informací jsou následně posílány do systému SIEM (*Security Information and Event Management*), který je zpracovává a je schopen v nich strojově hledat. Příkladem SIEMu může být ArcSight ESM (3, 64).

Řada výrobců aktivních prvků má implementovanou podporu pro zasílání síťových toků prostřednictvím protokolu NetFlow, sFlow, IPFIX, NetStream případně dalších variant. Nejpopulárnější variantou je NetFlow, které vyvinulo Cisco a na základě kterého byl postupně vyvinut otevřený standard pro síťové toky IPFIX. Nutno dodat, že někteří výrobci mají v aktivních prvcích implementovaný zmíněný protokol, ale ještě to neznamená, že výrobce implementoval veškerá pole datové struktury.

Existují také open source nástroje a utility pro generování Netflow. Příkladem může být *nprobe* nebo utility *nfdump* (ekvivalent *tcpdumpu*). Příkladem dodavatelů komerční platformy pro monitoring síťových toků je například společnost Flowmon, která dodává nejen sondy, ale i kolektor s uživatelským rozhraním pro analýzu a integrovaným systémem pro detekci anomálií (33, 65, 66).



Obrázek 2: Ukázka uživatelského rozhraní Netflow kolektoru od Flowmon Networks a.s. (Zdroj: 33)

1.4 Výchozí předpoklady

V rámci této práce není řešeno:

- fyzické zabezpečení komunikační infrastruktury ICS,
- metodika security hardeningu jednotlivých uzlů v síti, operačního systému pro běh IDS a jiných bezpečnostních aplikací,
- bezpečnostní architektura průmyslové sítě jako celku,
- systém pro centrální bezpečnostní monitoring (SIEM aj.), přestože některé části navrhovaného řešení pro tento systém figurují jako vstupy.

K bezpečnostnímu monitoringu mohou být využity i některé funkce zabudované v aktivních prvcích. Doporučenou metodou pro zabezpečení komunikace je *whitelisting* – veškerá komunikace nutná pro běh ICS bude v konfiguraci povolena a jakákoliv další komunikace bude vyhodnocena jako anomálie.

2 TEORETICKÁ VÝCHODISKA PRÁCE

V této kapitole jsou rozebrány jednotlivé pojmy, které jsou v diplomové práci použity. Kapitola obsahuje popis terminologie a pojmů v oblasti průmyslových řídících systémů, přehled pro počítačové sítě a komunikační protokoly a oblast zaměřenou na informační a kybernetickou bezpečnost.

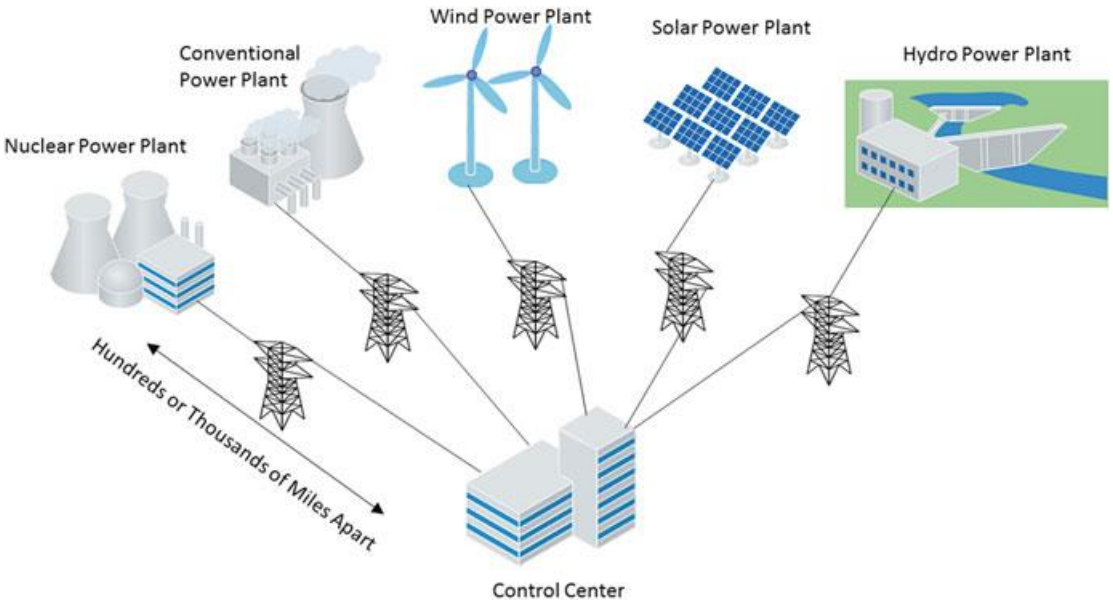
2.1 Průmyslové řídící systémy

Průmyslové řídící systémy (*Industrial Control System* – zkráceně ICS) označuje systémy skládající se z počítačů, elektrických a mechanických zařízení a manuálních činností vykonávaných člověkem – tento systém umožňuje buď plnou nebo částečnou automatizaci procesů pomocí techniky a to v různých oblastech – obecně výroba, vodohospodářstvím, doprava, energetika atd. Automatizace procesu a jeho řízení probíhá prostřednictvím digitálních zařízení – řídících prvků (jako PLC nebo RTU), a celkový monitoring a dálková kontrola je umožněna pomocí speciálních informačních systémů (SCADA systémy) (1).

Průmyslové řídící systémy bývají zpravidla postaveny na vlastní ICT infrastruktuře oddělené od té běžné komerční. Probíhá v nich odlišná komunikace a platí u nich jiná pravidla síťového provozu i bezpečnosti. Oproti komerčnímu ICT, kde je na prvním místě důvěrnost dat a informací, v systémech ICS je na prvním místě dostupnost řízeného procesu. V případě ICS je také kladen velký důraz na okamžitou odezvu, nutnou redundanci a značně delší očekávanou životnost jednotlivých komponent systému (5).

2.1.1 SCADA

SCADA (*Supervisory Control And Data Acquisition*) je speciální druh informačního systému, který umožňuje v reálném čase dálkově sbírat data, monitorovat a řídit proces v ICS (nehledě na jeho fyzické umístění) z jednoho nebo více řídících center.



Obrázek 3: ICS řízené z řídícího centra pomocí SCADA systému. (Zdroj: 1, s. 25)

Lokální řídící prvky v jednotlivých lokalitách (RTU, PLC, IED) komunikují prostřednictvím počítačové sítě s řídícím centrem. Komunikace probíhá pomocí protokolů, které jsou specifické pro průmyslové prostředí (DNP3, Ethernet/IP, Modbus, ICCP, IEC 60870-5-104). SCADA sama o sobě může být software provozovaný na jednom nebo více serverech se standardním operačním systémem (Windows, Linux), případně běžet na proprietárním OS. Pomocí SCADA systému lze nejen řídit a monitorovat, ale i vizualizovat samotný proces (5).

2.1.2 PLC

Programovatelná řídící jednotka (*Programmable Logic Controller* – PLC) je elektronické zařízení ovládané mikroprocesorem, které je schopno číst vstupní signály, spouštět na základě těchto vstupů (nebo případně dálkově zaslaných povelů řídícího systému) naprogramované instrukce a vysílat výstupní signály, spínat relé nebo regulovat akční členy ICS. Toto zařízení představuje hranici mezi digitálním a fyzickým prostředím. Často je dimenzované tak, aby odolávalo nepříznivým vlivům prostředí (vibrace, teplota, prašnost, vlhko, elektromagnetické rušení atd.). Moderní PLC mohou být modulární a umožňovat přizpůsobenou hardwarovou konfiguraci nebo rychlou výměnu poškozených modulů (1).

PLC má vždy napájecí jednotku, CPU, komunikační rozhraní (RS-232, Ethernet, …) a analogové nebo digitální vstupy a výstupy. Uvnitř PLC běží operační systém pro práci v reálném čase (*Real-Time OS* – RTOS), který může být postavený na mikro-kernelu odvozeného od Unixových operačních systémů. Na tomto operačním systému běží řídící smyčka, ve které dochází ke čtení vstupů, vykonávání instrukcí a zápisu do výstupů, to vše v řádech milisekund. Instrukce mohou být naprogramované proprietárním jazykem, případně mohou být naprogramovány ve standardizovaných jazycích (např. dle normy IEC 61131-3) (1).

Obrázek 4: Ukázky PLC – Siemens SIMATIC S7-300 (vpravo) a ABB AC500 (vlevo). (Zdroj: 6, 7)

2.1.3 RTU

RTU (*Remote Terminal Unit*) je elektronické zařízení obsahující mikroprocesor a určené pro instalaci v náročném prostředí. Obvykle se používají dva typy – *station* a *field* RTU. Field RTU v pravidelných intervalech sbírá signály ze senzorických zařízení a řídících členů instalovaných v terénu a na jejich základě vykonává instrukce. Field RTU je rozhraní mezi řídícími prvky / senzory instalovanými v terénu a Station RTU. Station RTU jsou také instalovány v odlehlých lokalitách a přijímají jak signály z field RTU, tak z nadřazeného řídícího centra. Station a Field RTU mohou být jedním zařízením (1).

RTU se skládá z napájecího zdroje, CPU a analogových nebo digitálních I/O modulů. Stejně jako PLC je hranicí mezi digitálním a fyzickým prostředím a je určené k řízení procesu. RTU mají často podobné schopnosti jako PLC a postupně implementují stejné programovací jazyky jako se používají k programování PLC. RTU může umět komunikaci s řídícím centrem přes WAN sítě prostřednictvím IP sítí, GPRS, mikrovln aj. a na toto centrum může průběžně nebo v pravidelných intervalech přeposílat sesbíraná data (1).

Obrázek 5: Ukázky RTU ABB řady 500. (Zdroj: 8)

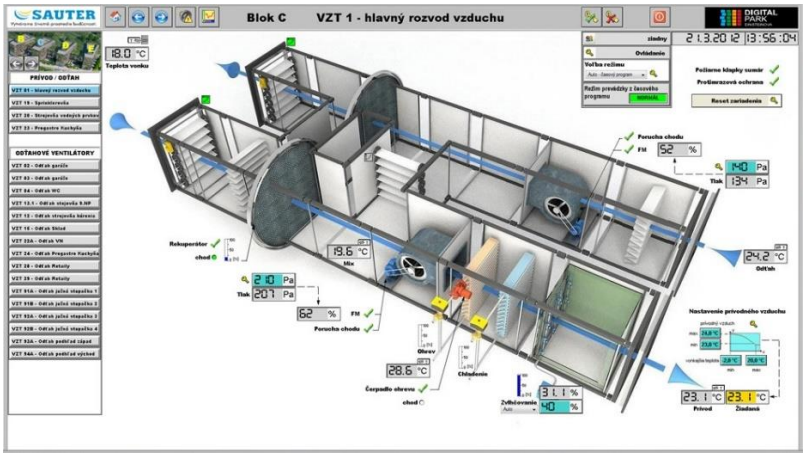
2.1.4 IED

IED (*Intelligent Electronic Device*) je elektronické zařízení obsahující jeden nebo více mikroprocesorů, které umí vysílat nebo přijímat z externího zdroje data a povely. Může být ovládáno řídící jednotkou nebo aplikačním serverem, a to buď přímo po síti Ethernet nebo prostřednictvím RTU. IED je často označované jako digitální ochranné relé. Vykonává pět funkcí: ochrannou, řídící, monitorovací, měřící a komunikační. Různé druhy IED od různých výrobců se z hlediska funkcí mohou lišit. Taková relé se využívají například v rozvodnách energetické distribuční soustavy, kde tyto zařízení detekují poruchy, zemní spoje, fázové posuny, přepětí atd. Tyto zařízení obvykle umožňují ovládání dálkové, ale i lokální pomocí zobrazovacího displeje na čelním panelu (1).

Obrázek 6: Ochranné relé od Schweitzer Engineering Laboratories. (Zdroj: 9)

2.1.5 HMI

HMI (*Human-Machine Interface*) je softwarová aplikace, pomocí které operátor může získat okamžitý přehled a grafickou vizualizaci řízeného procesu. HMI může běžet na různých platformách – dispečerských pracovních stanicích se standardním operačním systémem, tabletech, smartphonech nebo dedikovaných zobrazovacích terminálech. HMI také může být součástí SCADA systému. Tato aplikace může také umožňovat operátorovi přímé ovládání procesu. HMI obvykle graficky zobrazuje model ovládaného průmyslového řídícího systému včetně hodnot jednotlivých datových bodů, poplašných zpráv atd. (1).



Obrázek 7: Ukázka HMI systému SCADA Reliance běžícím na OS Windows. (Zdroj: 10)

2.2 Počítačové sítě

Počítačová síť je technologická infrastruktura, která umožňuje realizovat komunikaci a přenášet data mezi připojenými koncovými uzly. Fyzicky se vždy skládá z aktivních prvků (switche, routery, firewally, modemy atd.) a pasivní vrstvy (metalická kabeláž, optická vlákna, patch panely, optické vany, konektory) (11).

Za jeden ze základních aktivních prvků lze považovat switch, což je elektronické zařízení, které se v sítích využívá k doručování zpráv (rámců – PDU protokolu Ethernet) mezi jednotlivými připojenými uzly (protože je to prvek, který operuje na úrovni 2. vrstvy ISO OSI, lze jej označit jako L2 switch, existují ale i L3 switche podporující směrování v globální síti) (11).

Pravidla komunikace v počítačových sítích jsou obecně definována množinou komunikačních protokolů. Komunikační protokoly mohou definovat algoritmus výměny zpráv, zabezpečení přenášených dat, informovat o událostech a stavech v počítačové síti atd. Každý protokol definuje svou jednotku přenosu (PDU – *Protocol Data Unit*) a její přesnou strukturu. Pomocí skupiny protokolů lze definovat celou síťovou architekturu (viz. TCP/IP níže) (11).

Protože je tato práce zaměřena především na softwarovou stránku počítačových sítí, nebudou v teoretickém rozboru popsány síťové topologie, prvky strukturované kabeláže komunikační infrastruktury – teoretický rozbor se bude primárně věnovat komunikačním protokolům relevantním k předmětu práce.

2.2.1 Referenční model ISO OSI

Softwarové aplikace běžící na operačních systémech jednotlivých koncových uzlů zpravidla nepřenášejí data po síti přímo, ale prostřednictvím síťových služeb, které jsou naprogramované v operačním systému dané stanice. Síťové služby jsou dekomponovány do samostatných funkčních celků – hierarchicky seřazených vrstev, kdy nadřazená vrstva

vždy využívá služby poskytované nižší vrstvou. Dělení na jednotlivé vrstvy je zavedeno především z důvodu snadnějšího vývoje řešení a modularity (11).

O standardizaci řešení pro síťovou architekturu se pokusila organizace ISO, která vydala referenční model ISO OSI (*Open Systems Interconnection*). Přestože původním záměrem bylo vymyslet celou síťovou architekturu, ISO OSI bylo navržené velmi robustně a těžkopádně, proto se dnes v praxi využívá minimálně (výjimka např. viz MMS níže) a používá se spíš pro popis celkové logiky fungování síťových služeb (11).

ISO OSI definuje celkem 7 vrstev:

1. **Fyzická** (*Physical Layer*) – na této vrstvě jsou vysílány jednotlivé bity prostřednictvím komunikační infrastruktury a síťové karty stanice.
2. **Linková** (*Data Link Layer*) – obecnou jednotkou přenosu této vrstvy je rámec (frame). Úkolem této vrstvy je doručování rámců v dosahu přenosového média. Využívá se lokální adresace.
3. **Síťová** (*Network Layer*) – obecnou jednotkou přenosu je paket. Cílem této vrstvy je doručovat pakety mezi jednotlivými sítěmi. Využívá globální adresaci.
4. **Transportní** (*Transport Layer*) – obecnou jednotkou přenosu je datagram. Cílem je doručení dat konkrétnímu procesu na cílovém uzlu.
5. **Relační** (*Session Layer*) – zabezpečuje navázání spojení mezi aplikacemi na rozličných uzlech a výměnu dat mezi nimi.
6. **Prezentační** (*Presentation Layer*) – cílem této vrstvy je zabezpečení a transformace dat do podoby, se kterou může pracovat aplikační vrstva.
7. **Aplikační** (*Application Layer*) – tato vrstva má obsahovat standardizované části a moduly jednotlivých aplikací na cílové stanici (11).

2.2.2 Architektura TCP/IP

Architektura TCP/IP je dominantní architekturou v moderních počítačových sítích. Jedná se o otevřenou architekturu, která definuje základní protokoly pro komunikaci. Snaží se být kompatibilní a stavět na existujících řešeních. Oproti ISO OSI definuje pouze 4 vrstvy:

1. vrstva síťového rozhraní,
2. síťová – IP protokol,
3. transportní – TCP a UDP protokol,
4. aplikační vrstva (11).

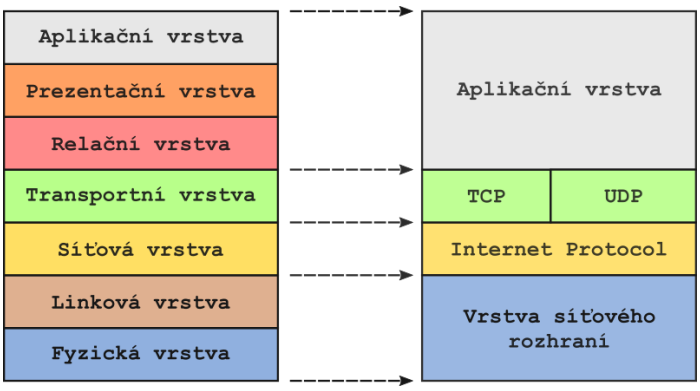
TCP/IP pro vrstvu síťového rozhraní nedefinuje žádné protokoly, architektura je univerzální a nezávislá na fyzických a linkových technologiích. TCP/IP nemá relační a prezentační vrstvu. Funkce těchto vrstev si musí zajišťovat samy aplikace, případně musí mít podporu v konkrétním aplikačním protokolu. Aplikačních protokolů fungujících na TCP/IP existuje celá řada (např. HTTP, DNS, TELNET, FTP, POP3, SMTP, SIP, SSH, IRC, RPC atd.). Základními definovanými protokoly architektury TCP/IP jsou IP, TCP a UDP (11).

IP protokol (*the Internet Protocol*) je protokol pracující na úrovní síťové vrstvy ISO OSI. Jeho účelem je zajistit přenos dat mezi připojenými uzly v globální síti (Internetu). Správné označení jednotky přenosu IP protokolu je datagram (jak je definováno v jeho RFC, přestože zažitější označení je paket). IP protokol podporuje globální adresaci pomocí IP adres, kdy adresa zdroje a cíle je obsažena v hlavičce datagramu (*headeru*). Adresa je 32 bitové číslo, které se u 4. verze protokolu standardně zapisuje ve formátu 4 dekadických čísel oddělených tečkou (např. *192.168.0.1*). IP protokol zajišťuje nespolehlivý přenos bez navazování spojení – spolehlivost a navázání spojení zajišťuje TCP (11).

TCP protokol (*Transmission Control Protocol*) je protokol pracující na úrovni transportní vrstvy ISO OSI. Pomocí něj lze zajistit spolehlivý přenos dat mezi procesy běžícími na dvou komunikujících uzlech. Je schopen ošetřovat ztráty dat, chyby v přenosu, správné pořadí doručení nebo duplicity. Vysoká přenosová režie má značný dopad na rychlost samotného přenosu dat. Jednotka přenosu je datagram. K adresování mezi procesy využívá tzv. porty (16 bitové číslo) (11).

UDP protokol (*User Datagram Protocol*) je protokol pracující na úrovni transportní vrstvy ISO OSI. Jednotka přenosu se taktéž označuje jako datagram a k adresaci využívá porty stejně jako TCP. Je to v porovnání s ním značně jednodušší protokol. Je vhodný pro datové

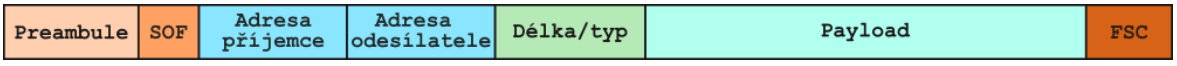
streamy (např. přenos videa) a přenosy dat, u nichž je rychlost přenosu nadřazena kvalitě doručení (11).



Obrázek 8: Srovnání ISO OSI a TCP/IP. (Zdroj: Vlastní zpracování dle: 11)

2.2.3 Ethernet

Ethernet je komunikační protokol pracující na úrovni linkové vrstvy ISO OSI. Řeší doručovaní dat v prostředí lokální sítě. Základní jednotkou přenosu je ethernetový rámec (frame). Existují dva základní typy Ethernetu – *klasický* Ethernet a *switched* Ethernet. Klasický Ethernet je převážně historická záležitost, která řeší problémy kolizí u vícenásobného přístupu k přenosovému médiu. Dnes se využívá výhradně switched Ethernet, kdy jsou jednotlivé koncové uzly připojené ke switchi, prostřednictvím kterého spolu komunikují. Existuje několik verzí Ethernetu a struktury jeho rámce. Pro potřeby práce nám poslouží obecná struktura dle IEEE 802.3 na následujícím obrázku (4).



Obrázek 9: Struktura rámce protokolu Ethernet. (Zdroj: Vlastní zpracování dle: 11)

1. **Preambule** a **SOF** (*Start Of Frame*) – 7+1 oktetů za účelem synchronizace příjemce. V prvních sedmi oktetech se střídá 1 a 0 (1010 1010) a následuje oktet, v kterém poslední bit nastavený na 1 označuje začátek rámce (1010 101**1**),

2. **Adresa příjemce** a **Adresa odesílatele** – každá adresa je dlouhá 6 oktetů (48 bitové číslo). Jedná se o tzv. MAC adresu. Každá vyrobená síťová karta má unikátní MAC adresu (např. 8C:EC:4B:0F:D8:38).
3. **Délka / Typ** – pole má délku dva oktety. Může obsahovat buď typ přenášených dat (např. 0x0800 označuje ethernetový rámec obsahující IP paket), nebo délku. V případě nižší hodnoty než 0x0600 se jedná o délku a v případě vyšší se jedná o typ.
4. **Payload** – obsahuje samotná přenášená data (v nich může být zapouzdřený protokol vyšší vrstvy). Délka je v rozmezí 46-1500 oktetů a v případě dat o menší velikosti než 46 oktetů se doplní o padding (nulové oktety).
5. **FSC** (*Frame CheckSum*) – jedná se o 32-bitový CRC kontrolní součet k ověření zachované integrity rámce (4).

Rámce lze posílat jedinému příjemci (tzv. *unicast*), skupině příjemců (tzv. *multicast*) anebo všem připojeným uzlům v lokální síti (tzv. *broadcast*). V sítích postavených na Ethernetu také platí (až na některé výjimky), že vždy pouze jedna MAC adresa smí náležet jedné síťové kartě. Oproti tomu v IP sítích může mít jedno zařízení více IP adres anebo více zařízení sdílet jednu IP adresu (4).

K překladu MAC adres na IP adresy se využívá protokol **ARP protokol** (*Address Resolution Protocol*). Ve zkratce ARP funguje následovně – vysílající má k dispozici IP adresu, ale neví, jaké MAC adrese patří – proto vyšle broadcastový rámec, kde se dotáže všech uzlů, komu náleží ona IP adresa. Následně uzel, kterému hledaná IP adresa náleží, odešle odpověď. Každý komunikující uzel má svůj ARP cache, kde jsou uloženy záznamy MAC-IP adres (4).

2.2.4 Průmyslové komunikační protokoly

Přestože existuje celá řada průmyslových protokolů, v této podkapitole následuje pouze popis relevantních protokolů užívaných k řízení distribuční soustavy energetické sítě – protokoly GOOSE (IEC 61850), MMS (ISO/IEC 9506) a protokol IEC 60870-5-104.

IEC 61850 definuje obecnou hierarchickou datovou strukturu pro operace a datové atributy v rámci distribuční rozvodny. Dle tohoto modelu rozlišujeme:

- **Fyzické zařízení** (konkrétní IED s IP a MAC adresou připojenou k LAN).
- **Logické zařízení** (*Logical Device*) – agreguje funkce a datové atributy z několika fyzických zařízení do jednoho logického. Obsahuje jeden nebo více logických uzlů.
- **Logický uzel** (*Logical Nodes*) – dle standardu nejmenší entita schopná vyměňovat data. Na základě funkcí jsou členěny do skupin (např. měření a správa, automatické řízení).
- **Datový objekt** (*Data Objects*) – každý logický uzel obsahuje datové objekty reprezentující aplikační objekty rozvodny a tyto objekty mají unikátní jméno. Jsou seskupeny do obecných kategorií (nastavení, řízení, měření, status, popis). Datové objekty mají své jednotlivé **atributy** s konkrétními hodnotami (12).

Komunikační model této normy rozlišuje dva základní druhy komunikace:

- **klient / server** – určený pro vyčítání dat z IED a přenášení větších datových objemů, které nejsou z hlediska času doručení kritické (MMS protokol),
- **peer-to-peer publisher / subscriber** – pro služby *Generic Substation Event* určené pro rychlou komunikaci mezi jednotlivými IED (GOOSE protokol) a periodické vysílání naměřených hodnot (SMV protokol) (12).

GOOSE (*Generic Object-Oriented Substation Event*) je protokol určený pro rychlý přenos událostí a příkazů v rámci LAN distribuční rozvodny, např. k rychlé a spolehlivé výměně událostí, dat a příkazů mezi jednotlivými IED. Využívá broadcastové a multicastové adresy, které vysílající (*publisher*) zasílá všem příjemcům (*subscribers*). Je to aplikační protokol využívající pro přenos linkový protokol (lze jej ale přenášet i přes IP a UDP). Standardně je přenášený ethernetovým rámcem dle IEEE 802.3, volitelně může obsahovat VLAN tag (*Virtual LAN*), HSR tag anebo PRP tag.

GOOSE má vyhrazená vlastní čísla typu v ethernetovém rámci (*Ethertype* 0x88B8-0x88BA) a vysílá na multicastových adresách 01:0C:CD:01:**XX**:**XX**. Má vždy hlavičku a datovou část. Pro účel této práce není potřeba uvádět detail jeho struktury a zapouzdření v rámci (12).

MMS (*Manufacturing Message Specification*) je komunikační protokol, který umožňuje modelovat reálná zařízení a jejich funkce, vyměňovat data o zařízení a řízeném procesu. Je definovaný dle ISO/IEC 9506 a v normě IEC 61850 je popsán způsob, jakým je obecný datový model mapovaný na tento protokol. MMS využívá komunikační model klient-server, kdy klientem je monitorovací aplikace nebo dohledový systém a serverem je zařízení nebo aplikace obsahující tzv. *Virtual Manufacturing Device* společně s jeho objekty. MMS je objektově orientovaný protokol (rozlišuje třídy s jejich instancemi a metodami).

MMS nedefinuje adresaci klienta a serveru (využívá se IP adresace) a protokol je zapouzdřen uvnitř TCP datagramu (standardně port 102). MMS je ale až na úrovni aplikační vrstvy ISO OSI a společně s ním jsou v TCP datagramu zapouzdřeny i protokoly transportní vrstvy (TPKT, COTP), relační vrstvy (*OSI Session-Oriented Transport Protocol*), prezentační vrstvy (*OSI Connection Oriented Presentation*) a aplikační vrstvy ISO OSI (*Association Control Service Element*) (12).

IEC 60870-5-104 je aplikační protokol, který se využívá k přenosu telemetrických dat a dálkovému ovládání v průmyslových řídících sítích energetické distribuční soustavy. IEC 60870-5-104 je ve skutečnosti norma pro přenos modifikovaného protokolu IEC 60870-5-101 v architektuře TCP/IP – zapouzdřuje PDU protokolu do datové části TCP datagramu. Klient (např. dohledový systém) následně vysílá příkazy na IEC 60870-5-104 server (na port 2404) (13).

Struktura PDU protokolu IEC 60870-5-104 (*APDU – Application Protocol Data Unit*) je možno rozdělit na dvě části – APCI (*Application Protocol Control Information*) a ASDU (*Application Service Data Unit*). APCI vždy na začátek obsahuje oktet 0x68, délku celého APDU a 4 oktety pro řídící parametry (*Control Fields*). Ty podmíněně obsahují sekvenční čísla a v závislosti na těchto parametrech lze také určit typ APCI:

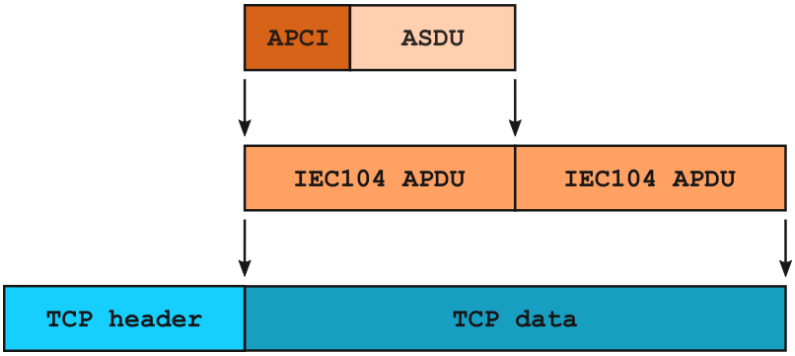
- **I-frame** (*Information Transfer Format*) – tento typ APCI je využíván k přenosu informací mezi řídící a řízenou stanicí. Je proměnlivé délky a vždy obsahuje ASDU.

- **S-frame** (*Numbered Supervisory Functions*) – využívá se k posílání číslovaných dohledových funkcí, např. řídící stanice pošle řízené zpět potvrzovací sekvenční číslo, v případě že se data posílají pouze v jednom směru. Má pevnou délku 6 oktetů.
- **U-frame** (*Unnumbered Control Functions*) – využívá se k posílání nečíslovaných dohledových funkcí, konkrétně tří – *Start Data Transfer*, *Stop Data Transfer* a *Test Frame*. Např. *Start Data Transfer* se posílá při navázání spojení s řízenou stanicí; má pevnou délku 6 oktetů (13).

ASDU obsažené v I-frame obsahuje samotné hodnoty jednotlivých elementů datových objektů řízené stanice. Následuje přehled vybraných polí ASDU:

- *Type identification* – toto pole určuje typ ASDU, tj. co přenáší, resp. vykonává (např. „*Bit string of 32 bit with time tag*“ označuje ASDU které přenáší 32-bitový textový řetězec s časovou známkou).
- *Number of objects/elements* – určuje počet dotčených informačních objektů / elementů.
- *Cause of Transmission* – toto pole určuje směrování zprávy, resp. jak s ní naložit. Určuje k jakému programu bude daná zpráva přiřazena. Každý typ ASDU má své odpovídající hodnoty COT.
- *Common Address of ASDU* (COA) – obvykle se jedná o adresu konkrétní stanice, lze ale využít tuto adresaci k rozčlenění stanic na samostatné logické jednotky (13).

Konkrétní nastavení jednotlivých polí protokolu se může lišit v závislosti na aplikaci a implementaci. Uvnitř TCP datagramu se může nacházet několik APDU sériově za sebou, kdy každá APDU může mít zapouzdřeno ASDU jiného typu. Tuto situaci lépe znázorňuje následující obrázek:



Obrázek 10: Zapouzdření dvou APDU. (Zdroj: Vlastní tvorba)

2.2.5 Syslog

Syslog je protokol pro přenos systémových událostí po síti. Tyto zprávy mohou být následně uloženy např. z důvodů provozního nebo bezpečnostního monitoringu. Protokol je definovaný dle volných norem RFC. Existují 2 verze – RFC 3164 pro starší BSD syslog a RFC 5424 pro novější verzi (14, 15).

Starší BSD Syslog obsahuje tři části:

- **PRI** (*Priority Value*) – jedná se o číslo, pomocí kterého se určí závažnost a zařazení (*priority* a *facility*) dané zprávy. Existuje 8 úrovní závažnosti:
  - [0] emergency,
  - [1] alert,
  - [2] critical,
  - [3] error,
  - [4] warning,
  - [5] notice,
  - [6] informational,
  - [7] debug.

  Jednotlivé zařazení mají přiřazena čísla v rozmezí 0-23 (např. kernel zprávy [0], security / authorization [10]). Číslo PRI je zapsáno ve formátu „<123>“. Kód facility je vždy vynásoben 8 a k němu je přičten kód závažnosti.

- Hlavička (*Header*) – hlavička obsahuje časovou známku a buď IP adresu nebo hostname vysílající stanice.
- Zpráva (*Message*) – obsahuje samotnou zalogovanou zprávu. Ta se skládá z tagu (názvu programu nebo procesu logující zprávu) a obsahu (14).

Novější verze syslogu přidává další pole (verzi syslogu, APP-NAME, PROCID, MSGID, etc.). Navzdory tomu vždy záleží na výrobci zařízení, jaká z těchto polí využije v implementaci syslogového agenta. V návrhové části je k syslogovým záznamům uvedena zpráva bez PRI čísla (15).

2.2.6 NetFlow a IPFIX

NetFlow je aplikační protokol vyvinutý společností Cisco, který umožňuje uživateli agregovat statistiky o proběhlých komunikačních tocích v síti. NetFlow statistiky může generovat síťový aktivní prvek (switch, router), případně pro jeho generování může být využita samostatná sonda. O síťových tocích lze sledovat parametry jako IP adresy, počty přenesených paketů, počet přenesených bytů, časové známky, ToS (*Type of Service*), porty vstupní a výstupní rozhraní atd. (16).

NetFlow se zapouzdřuje do UDP datagramu, uvnitř kterého je vždy hlavička a jednotlivé sady toků (tzv. *FlowSets*). Tyto sady se dále dělí na *data*, *template* a *options flowsets*. Uvnitř těchto sad jsou záznamy o proběhlé komunikaci. Existují různé verze protokolu, nejnovější je v současnosti verze 9. Tuto verzi rozšiřuje IETF standard pro IPFIX, který je následníkem NetFlow (16, 17).

2.3 Kybernetická a informační bezpečnost

Informační bezpečnost (bezpečnost informací) řeší ochranu informací a jejich dostupnost. Úzce souvisí s bezpečností organizace a její IS/ICT infrastruktury. Bezpečnost organizace souvisí se zajištěním bezpečnosti objektů a majetku dané organizace, do kterého spadají i aktiva IS/ICT. Informační bezpečnost se zabývá i bezpečností informací v nedigitální

podobě. Informační bezpečnost má na rozdíl od kybernetické bezpečnosti pevně dané hranice – perimetr (3).

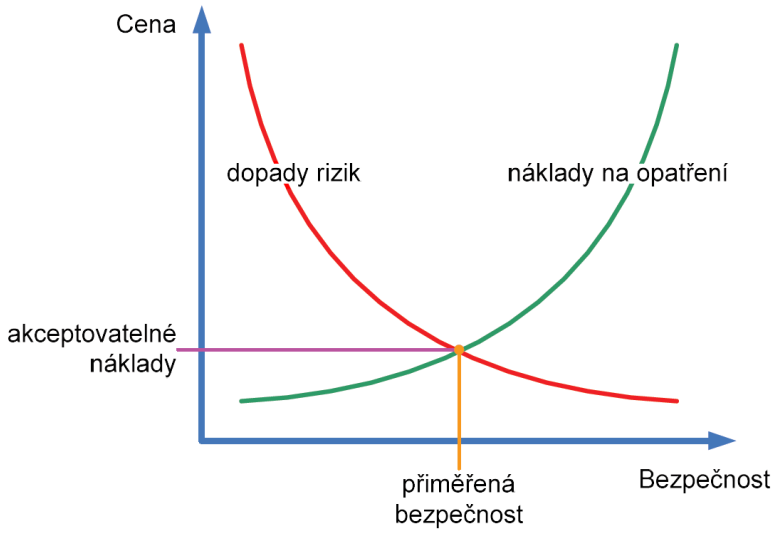
Kybernetická bezpečnost je souborem opatření, která pomáhají chránit systémy IS/ICT před kybernetickým útokem nebo neautorizovaným přístupem. Jedná se o souhrn právních, organizačních, technických a vzdělávacích prostředků k zajištění ochran kybernetického prostoru. Kybernetická bezpečnost řeší bezpečnost v kyberprostoru (2).

Kyberprostor je digitální prostředí umožňující vznik, zpracování a výměnu informací vytvořených jednotlivými informačními systémy a službami. Je postaven na prvcích informačních a komunikačních technologií, které vytvářejí celosvětovou počítačovou síť a v které spolu připojené počítačové systémy komunikují a provádějí interakci. Lze ho označit za virtuální realitu postavenou na reálných technologických prostředcích v našem světě. Je to otevřené a decentralizované prostředí, které nemá pevně vymezené hranice (2).

Kybernetickou bezpečnostní událost definuje Zákon o Kybernetické Bezpečnosti (§7, odst. 1) jako „*událost, která může způsobit narušení bezpečnosti informací v informačních systémech nebo narušení bezpečnosti služeb anebo bezpečnosti a integrity sítí elektronických komunikací*“. Jedná se o negativní událost ohrožující informační aktivum, která zatím nemá žádný negativní dopad (2).

Kybernetický bezpečnostní incident je dle Zákona o Kybernetické Bezpečnosti (§7, odst. 2) „*narušení bezpečnosti informací v informačních systémech nebo narušení bezpečnosti služeb anebo bezpečnosti a integrity sítí elektronických komunikací v důsledku kybernetické bezpečnostní události*“. U incidentu na rozdíl od události už dochází k reálnému dopadu na informační aktiva dané organizace (2).

Abychom zabránili událostem a dopadům incidentů na naše informační aktiva, je třeba zavést odpovídající opatření. U nich ale musíme dbát na přiměřenost – usilujeme o **přiměřenou bezpečnost**. Objem vynaložených prostředků a úsilí musí být poplatný hodnotě aktiv, která se snažíme ochránit.



Obrázek 11: Grafická interpretace přiměřené bezpečnosti. (Zdroj: 3, s. 36)

2.3.1 CIA triáda

CIA triáda označuje základní vlastnosti a zároveň cíle kybernetické bezpečnosti daného informačního aktiva. Má tři parametry – důvěrnost, integritu a dostupnost.

Důvěrnost (***C****onfidentiality*) označuje fakt, že k datům, informacím nebo případně ICT má přístup pouze oprávněný subjekt. V rámci organizace je vhodné provádět klasifikaci informací. Existují různé stupně klasifikace – příkladem může být tzv. *Traffic Light Protocol* (TLP). TLP klasifikuje informace na 4 stupně – **bílá** (zveřejnění není nijak omezeno), **zelená** (zveřejnění je omezeno na komunitu), **oranžová** (zveřejnění možné pouze pro organizaci účastníků) a **červená** (zveřejnění pouze pro samotné účastníky) (2).

Integrita (***I****ntegrity*) je samotná správnost a úplnost dané informace, pokud je zajištěna integrita informací, dat nebo ICT systémů, pouze oprávněná osoba může tato data / informace změnit, případně změnit konfiguraci ICT systému (2).

Dostupnost (***A****vailability*) lze označit jako garanci možnosti přístupu k informačnímu aktivu v okamžiku potřeby, případně procesu, které informační aktiva podporují. Pokud dojde k zničení informací, dat nebo systému ICT, jedná se o narušení jejich dostupnosti (2).

U ICS systémů lze na dostupnost také pohlížet z hlediska spolehlivosti (bezporuchovosti) systémových komponent. U těchto komponent posuzujeme dva základní parametry:

- **Střední meziporuchovou dobu** (*Mean Time Between Failures* – MTBF) – usilujeme o to, aby byla co nejdelší. Jedná se o statistickou veličinu, která hodnotí spolehlivost daného zařízení. Obvykle se udává v letech.
- **Střední dobu do obnovení** (*Mean Time To Restore* – MTTR) – usilujeme o to, aby byla co nejkratší. Doba nahrazení / opravení dané komponenty a návratu k původní činnosti subjektu. Obvykle se udává v hodinách (18).

Na základě těchto parametrů lze spočítat celkovou dostupnost dané systémové komponenty dle následujícího vzorce:

$$Dostupnost = \frac{MTBF}{MTBF + MTTR}$$

Hodnota dostupnosti bude limitovat k 1 (18).

2.3.2 Kritická informační infrastruktura a Zákon o Kybernetické Bezpečnosti

Kritická infrastruktura je definována v Zákoně o Krizovém Řízení. Jedná se o prvek infrastruktury státu, kdy narušení jeho funkce může mít zásadní negativní dopad na bezpečnost nebo ekonomiku státu, zdraví nebo zajištění základních životních potřeb lidí na jeho území. Její součástí je **kritická informační infrastruktura**, která označuje ICT prostředky a informační aktiva (2).

Kvůli vzrůstající závislosti společnosti na ICT vzniká i riziko zneužití těchto technologií nebo útoku na tyto technologie, což může vést ke značným škodám a mít dopad na subjekty, které s těmito technologiemi pracují. Je tedy cílem tyto technologie ochránit, zvlášť ty, které jsou součástí kritické informační infrastruktury státu. Za tímto účelem by vydán **Zákon** č. 181/2014 Sb., **o Kybernetické Bezpečnosti** (novelizovaný č. 205/2017 Sb.). Jeho cílem je zajištění bezpečného fungování české informační společnosti. (2).

2.3.3 Přehled dalších vybraných pojmů

CERT (*Computer Emergency Response Team*) a **CSIRT** (*Computer Security Incident Response Team*) jsou týmy, jejichž úlohou je řešit problematiku kybernetické bezpečnosti. Přestože se jedná o dvě různé zkratky (označení CERT je pod ochrannou známkou), mohou vykonávat v rámci organizace / komunity stejné činnosti. Jedná se o tým, který má ve svém poli působnosti odpovědnost za řešení bezpečnostních incidentů a hrozeb. Pro ostatní subjekty jsou týmem, na který se mohou obrátit s řešením incidentů, spolupráci v oblasti informační a kybernetické bezpečnosti atd. (2).

Řízení kontinuity (*Business Continuity Management* – BCM) je procesem v organizaci, jehož cílem je identifikovat klíčové systémy a procesy, a následně zavedení procesů, postupů a opatření, které zajistí jejich kontinuitu nebo rychlou obnovu, tak aby na předem definované úrovni dál organizace fungovala a plnila své úlohy. Součástí procesu BCM je i příprava *disaster recovery* plánu (2).

IDS (*Intrusion Detection System*) jsou systémy, které na základě sledování síťové komunikace nebo aktivit a chování procesů v operačním systému identifikují případné pokusy o nežádoucí aktivitu a následně vyhlásí poplach. Mohou být instalované na koncovém uzlu – HIDS (*Host-Based IDS*) nebo síťové (*Network IDS*). **IPS** (*Intrusion Prevention System*) v porovnání s IDS je schopen aktivně zasáhnout nebo zabránit dané nežádoucí aktivitě (2)

DLP (*Data Loss Prevention*) je systém, který identifikuje, monitoruje a chrání data. To je prováděno na základě hloubkové kontroly obsahu a analýzy datových transakcí. DLP systém lze rozdělit na 3 části, dle toho, co chrání – ochrana **používaných dat** (*Data in use*), ochrana **dat v pohybu** (*Data in motion*) a ochrana **dat v klidu** (*Data at rest*). DLP mohou být **síťová** (sledují síťové komunikace, typicky HTTP, FTP, email) a *host-based* (běží jako klienti na koncových stanicíc). (3)

SIEM (*Security Information and Event Management*) je systém, který monitoruje a agreguje bezpečnostní informace, logy a síťové toky v dané infrastruktuře a provádí nad nimi korelaci,

je schopen tyto informace zobrazovat uživateli k intepretaci. Díky jeho výstupům lze odhalit potenciální hrozby nebo kybernetické bezpečnostní události v síti. SIEM může přijímat vstupy z log managementu (centralizované správě systémových událostí v IS/ICT infrastruktuře), IDS / IPS, DLP nebo dalších technologických prostředků kybernetické bezpečnosti (2).

3 VLASTNÍ NÁVRHY ŘEŠENÍ

Tato kapitola obsahuje vlastní návrh řešení. Navrhované řešení může být implementováno buď v plném rozsahu, nebo jen částečně, případně jej lze dle potřeby optimalizovat nebo doplnit.

Navrhované řešení se skládá ze dvou skupin opatření – **technologická** a **organizační**. Technologická opatření zahrnují návrh technologií pro monitoring síťového provozu, jejich konfiguraci a srovnání bezpečnostních funkcí vybraných switchů. Organizační opatření doplňují ta technologická o nutné firemní procesy, které je třeba v rámci společnosti implementovat, aby byl problém zabezpečení systému ICS komplexně vyřešen. Návrh dále obsahuje popis **fází implementace**, jejich dílčích kroků a **ekonomické zhodnocení** navrhovaného řešení.

Pro potřeby testování technologií byl poskytnut záznam síťového provozu ve formátu PCAP a virtuální server, na který byl pomocí mirroringu duplikován provoz lokálního řídícího systému z vybrané distribuční rozvodny.

3.1 Technologická opatření

Na základě analýzy síťového provozu bylo zjištěno, že v prostředí ICS jsou využívány jak běžné protokoly, se kterými se lze setkat i v komerčních IT sítích, tak průmyslové protokoly, a to především protokol **IEC 60870-5-104** a protokoly **GOOSE** a **MMS**. Protokoly MMS a IEC 60870-5-104 jsou oba aplikační protokoly využívající pro přenos TCP/IP, lze tedy na ně relativně snadno psát pravidla ve standardním otevřeném formátu, který využívají open source NIDS Snort a Suricata (stejně tak jako některá komerční řešení).

IEC 60870-5-104 komunikace probíhá mezi lokální řídící jednotkou a centrálním SCADA systémem (na rozdíl od MMS). V rámci návrhu budou pravidla definována pouze pro tento protokol. GOOSE je zapouzdřený v linkovém protokolu a pracuje v real-time režimu.

Na GOOSE *nelze ve výše zmíněném formátu definovat pravidla* ani využít Snort / Suricatu pro jeho monitoring.

Protože NIDS vyhlásí poplach jen v případě, že budou splněny všechny parametry jednoho pravidla, *je žádoucí v řešení zakomponovat i monitoring síťových toků*, který umožní detekci obecných anomálií síťového provozu.

Pro bezpečnost na úrovni protokolů linkové vrstvy musí být využito buď komerčních řešení, nebo postačí zajistit bezpečnost pomocí aktivních prvků. Pro dosažení maximální úrovně bezpečnosti je ideální kombinace obojího – je třeba zvážit cenu investice, míru rizika a závažnost dopadu. V rámci návrhu jsou srovnány možnosti bezpečnostních funkcí switchů pro průmyslové sítě od tří vybraných výrobců.

Po nasazení řešení do ostrého provozu musí být také zajištěno následující:

- Je třeba se ujistit, že mirroring příliš nezatěžuje aktivní prvky průmyslové sítě.
- Je třeba nastavit filtrování logů a adekvátní frekvenci zasílání síťových toků (a případných SNMP trapů). V opačném případě by data odesílaná na systém centrálního monitoringu zbytečně zahlcovala komunikační kanál.

Je třeba zvážit, jestli je vhodné využití open source technologií pro bezpečnostní monitoring. Open source technologie mají sice nulové pořizovací náklady, nicméně vyžadují větší režii z hlediska údržby, ladění a správy, obzvlášť pokud je technik odkázaný pouze na základní dokumentaci dostupnou online a k produktu není zakoupená žádná podpora. Výhodou je relativní možnost produkt jakkoliv uzpůsobit.

Specializované komerční NIDS jsou podstatně dražší, mají vyšší pořizovací náklady než standardní komerční řešení, mnohdy je třeba platit roční podporu a v případě potřeby uzpůsobit produkt je zákazník plně závislý na vůli dodavatele a možnostech smlouvy. Doplnit komerční řešení o open source nástroje poskytuje členům CERT/CSIRT více možností. Ideální se tedy jeví kombinace obojího, ale v návrhu budou zahrnuty pouze open source NIDS.

3.1.1 Pravidla Intrusion Detection Systems

Protože Snort 2 trpí na nedostatky popsané v analytické části a Snort 3 je ještě v Beta verzi, bude v návrhu využita Suricata jako hlavní detekční nástroj. Pravidla si může uživatel vytvořit sám nebo využít souborů pravidel distribuovaných online. Existují dva hlavní zdroje pravidel pro Suricatu – *Emerging Threats* a *Talos (dříve VRT)* (37, 34).

Oba zdroje poskytují knihovny pravidel zdarma i v placené verzi zaměřené čistě na komerční sítě a předpokládají přístup z LAN do sítě Internet. Přestože se v našem případě jedná o izolované průmyslové prostředí, lze využít minimálně neplacených verzí těchto knihoven.

Z oficiálních stránek Snortu lze stáhnout open source pravidla, po registraci lze stáhnout zdarma dodatečná pravidla. Placená sada obsahuje pravidla zveřejněná ihned v reakci na nové hrozby, malware a zranitelnosti, zatímco pravidla vyžadující pouze registraci jsou o 30 dní pozadu vzhledem k těm placeným. VRT pravidla jsou primárně určena pro Snort, a ještě k tomu vázaná na konkrétní verzi. Využití Snort pravidel by vyžadovalo ze strany uživatele určitých úprav. Některá klíčová slova pravidel Suricata interpretuje jiným způsobem, případně zavádí nová (27).

ET pravidla obsahují mimo jiné seznam veřejných IP adres, které jsou známy škodlivou aktivitou. Přestože je síť izolovaná, v případě, že bude v síti detekován pokus o komunikaci na kteroukoliv z takových IP adres, jedná se s velkou jistotou o potenciální kompromitaci zařízení malwarem. U takových pravidel pochopitelně nehrozí false positive.

Obecný formát pravidel

Pravidla mají jednoduchou strukturu. Příkladem může být následující pravidlo z ET sady:

```
drop tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"ET TROJAN Likely
Bot Nick in IRC (USA +..)"; flow: established, to_server; flowbits:
isset, is_proto_irc; content: "NICK "; pcre: "/NICK .*USA.*[0-
9]{3,}/i"; reference: url ,doc.emergingthreats.net/2008124;
classtype: trojan-activity; sid:2008124; rev:2;)
```

(Zdroj: 35)

Červenou barvou je označená akce, kterou má NIDS s packetem v případě platnosti pravidla vykonat. Protože v rámci návrhu bude používána Suricata pouze v režimu IDS a ne IPS, akce bude vždy *alert*. **Zelená** označuje protokol, zdrojovou IP adresu, zdrojový port, směr přenosu, cílovou IP adresu a cílový port. **Modrá** označuje všechny parametry pravidla.

Parametry se mohou vztahovat k obsahu paketu/datagramu, zprávě, kterou má IDS zapsat, kategorickému zařazení pravidla atd. Podrobnosti o možnostech konfigurace lze najít v oficiální dokumentaci Suricaty nebo Snorta (dokumentace Suricaty neobsahuje popis všech klíčových slov, která podporuje, proto je někdy nutné využít Snort dokumentace).

Pravidla jsou poměrně jednoduchá, což je jejich silnou a zároveň slabou stránkou. Vztahují se vždy ke konkrétnímu paketu/datagramu. Výhodou je, že se snadno píšou a je možno se v nich dobře orientovat. Na druhou stranu je obtížné pomocí základních klíčových slov sestavit pravidlo, které sepne na komplexnější scénář.

Komplexnější kontroly na úrovni jednoho paketu lze dosáhnout pomocí **Lua JIT** (*Just-In-Time*) scriptu – pravidlo spustí dodatečný script v jazyce Lua, kde proběhne podrobnější vyhodnocení obsahu. Pomocí těchto pravidel nelze provádět kontrolu několika paketů zároveň (např. celé jedno spojení).

Pravidla pro IEC 60870-5-104

V následující podkapitole jsou uvedeny navrhované metody, pomocí jakých lze provádět kontrolu nad protokolem IEC 60870-5-104. Ten je hlavním protokolem využívaným pro dálkové ovládání v zabezpečovaném průmyslovém prostředí. Veškerá pravidla byla testována na poskytnutém záznamu síťového provozu. Záznam byl manuálně načten do Suricaty verze 4.1.2, pomocí které byla testována níže uvedená pravidla.

Protokol standardně využívá TCP na portu 2404. Už jen na základě portu lze omezit komunikaci mezí lokálním IEC 60870-5-104 serverem a centrálním SCADA systémem (IEC 60870-5-104 klientem). Předpokládejme, že proměnná **$IEC104_CLIENT** označuje

všechny vzdálené řídící SCADA servery a proměnná **$IEC104_SERVER** označuje všechny dálkově ovládaná zařízení v lokální síti. Negace pomocí vykřičníku nám umožní napsat pravidlo, které vyhlásí poplach pro každý pokus o komunikaci z neznámého zařízení na lokální IEC 60870-5-104 server (RTU nebo PLC):

```
alert tcp !$IEC104_CLIENT any -> $IEC104_SERVER 2404 (msg:
"Suspicious IEC-104 communication."; classtype: bad-unknown; sid:
1040001; rev: 1;)
```

Každá přenosová jednotka protokolu **APDU** (*Application Protocol Data Unit*) obsahuje hlavičku **APCI** (*Application Protocol Control Information*) která začíná oktetem **0x68**. APDU mohou dále být tří typů – *I-frame*, *S-frame* a *U-frame*. U-frame je například poslán, pokud centrální SCADA server vydá povel „*Stop Data Transfer*“. Pomocí následujícího pravidla vyhlásí NIDS poplach při detekci každého takového rámce zaslaného na lokální IEC 60870-5-104 server (13):

```
alert tcp any any -> any 2404 (content:"|68 04 07|"; offset:0;
depth:3; msg:"Suspicious IEC-104 U-frame function: STOPDT  act.";
classtype: bad-unknown; sid: 1100404; rev: 1;)
```

Pomocí klíčového *content* lze označit konkrétní obsah v payloadu TCP datagramu. Slovo *offset* určuje číslo prvního kontrolovaného oktetu a *depth* určuje, kolik oktetů má být kontrolováno (36).

V znázorněném pravidle můžeme vidět oktet **0x68** označující začátek APDU, **0x04** označuje délku obsahu APDU (u U-frame a S-frame bude vždy 4) a **0x07** označí vždy U-frame funkci „*STOPDT act*“ (13).

U U-frame funkcí může řídící server vyslat *activation* funkci a v reakci na to řízený server odešle nazpět *confirmation* funkci. Pokud chceme vyhlásit poplach pro *activation* nebo *confirmation* zaslané v nesprávném směru, můžeme využít následujícího pravidla:

```
alert tcp any 2404 -> any any (content:"|68 04|"; offset:0; depth:
2; pcre: "/(\x43|\x13|\x07)/RA"; msg:"Suspicious IEC-104 U-frame
```

```
function sent from SLAVE to MASTER."; classtype: bad-unknown; sid:
1100406; rev: 1;)
```

Je možné definovat buď tři pravidla pro každou funkci zvlášť nebo využít klíčového slova *pcre* a s využitím regulárních výrazů definovat jedno pravidlo. Zvýrazněná červená část označuje regulární výraz, na základě kterého IDS vyhlásí poplach v případě, že na třetím oktetu APCI bude jedno ze tří hexadecimálních čísel. Protože je využití klíčového slova *pcre* výpočetně náročnější, je vhodné nejdřív použít dodatečný parametr (viz modře označená část).

I-frame jsou APDU která obsahují samotná data. Kromě APCI obsahují také **ASDU** (*Application Datagram Service Unit*). ASDU obsahuje řadu polí nad kterými lze provádět kontrolu, nicméně pro účely návrhu byly vybrány pole **TypeID** (*Type Identification*) a **COT** (*Cause of Transmission*) (13).

V současné době existuje 58 unikátních TypeID, nicméně v rámci konkrétního aplikace nemusí být využity všechny. 8bitové pole pro TypeID může nabývat hodnot 0-255, nicméně určité rozsahy čísel představují rezervy nebo jsou rezervovány v rámci normy (13).

Následující pravidlo nám umožňuje vyfiltrovat pomocí regulárního výrazu hodnoty TypeID 0-100 a 104:

```
alert tcp any any -> any any (content:"|68|"; offset:0; depth:1;
pcre: "/[\S\s]{5}([\x00-\x64]|[\x68])/RA"; msg:"Suspicious IEC-104
TypeID."; classtype: bad-unknown; sid: 1040628; rev: 1;)
```

COT pole obsahuje číslo v hodnotách 0-63. Protože pole nemá délku celého oktetu, nelze ke kontrole využít klíčových slov *content* nebo *pcre*. Namísto toho lze využít klíčového slova *byte_test*, které nám umožňuje provádět bitové operace nad vybraným oktetem. V rámci navrhovaného pravidla má *byte_test* čtyři vstupy, jejichž význam je následující:

1) první vstup udává počet testovaných oktetů,
2) druhý vstup určuje binární operaci, která je s oktetem prováděna,
3) třetí vstup udává hodnotu, proti které testujeme,

4) čtvrtý vstup udává číslo prvního oktetu, který je určen k testování (38, 39).

Následující pravidlo vyhlásí poplach při detekci APDU s hodnotou COT rovno 3:

```
alert tcp any any <> any 2404 (content:"|68|"; offset:0; depth:1;
byte_test:1,!&,1,2;  byte_test:1,!&,32,8; byte_test:1,!&,16,8;
byte_test:1,!&,8,8; byte_test:1,!&,4,8; byte_test:1,&,2,8;
byte_test:1,&,1,8; msg:"Suspicious IEC-104 ASDU COT: spont (3).";
classtype: bad-unknown; sid: 1040703; rev: 1;)
```

Protože je třeba kontrolovat individuální bity, bylo třeba použít *byte_test* celkem šestkrát. Dodatečně bylo vhodné se ujistit, že bude vždy kontrolován pouze IEC 60870-5-104 datagram obsahující *I-frame* (kvůli úspoře výpočetního výkonu). Proto byl využit ještě sedmý *byte_test* (modře). Symboly **&** a **!&** (červeně) jsou označení pro binární operaci AND a NOT AND, pomocí kterých bylo zjištěno, jestli je nebo není daný bit sepnutý (38).

COT je hodnota vázaná pouze na některá TypeID. COT rovno 1 je využito pouze u TypeID rovno 21. Následující pravidlo vyhlásí poplach u COT rovno 1, kde se TypeID nerovná 21:

```
alert tcp any any <> any 2404 (content:"|68|"; offset:0; depth:1;
byte_test:1,!&,1,2; pcre: !"/\x68[\S\s]{5}[\x15]/A";  byte_test:
1,!&,32,8; byte_test:1,!&,16,8; byte_test:1,!&,8,8; byte_test:
1,!&,4,8; byte_test:1,!&,2,8;byte_test:1,&,1,8; msg:"Suspicious
IEC-104 ASDU COT: pec/cyc (1)."; classtype: bad-unknown; sid:
1040701; rev: 1;)
```

Pomocí Snort/Suricata pravidel lze testovat jak na úrovni bytu i jednotlivých bitů, je ale třeba mít k dispozici specifikaci protokolu, případně *Proof of Concept* souvisejícího útoku. Jako specifikace pro IEC 60870-5-104 posloužil technický report Ing. Petra Matouška, Ph.D. (viz zdroj č. 13). Tvorba výše uvedených pravidel byla inspirována prací pana Y. Yanga a dalších (40).

Skupina Cisco Talos v roce 2016 vydala 33 pravidel určených pro tento protokol a na oficiálním blogu Snort NIDS je jejich popis, přestože jsou neveřejná. Nicméně kromě detekce TypeID a U-frame funkcí neobsahují dle zdroje téměř nic dalšího a jen kvůli těmto

pravidlům se nevyplatí pořizovat komerční sadu pravidel (nenacházejí se ani v sadě pravidel dostupné po registraci) (41).

Všechna výše uvedená pravidla nepočítají s několika APDU v jednom TCP datagramu. Jsou založena na předpokladu, že všechna APDU v jednom TCP datagramu mají stejné vlastnosti (TypeID, COT). Komplexnější kontrolu by bylo potenciálně možno vyřešit pomocí Lua scriptu, nicméně na úkor vyšší výpočetní náročnosti.

Před konfigurací, nasazením a testováním IDS a pravidel je třeba konzultovat se správcem, případně dodavatelem aplikace, jaké parametry protokolu jsou v rámci komunikace využívány.

Další pravidla

Obdobným způsobem lze psát pravidla i pro další aplikační protokoly jako je MMS. Opět je nutným předpokladem dostupná specifikace protokolu a informace o tom, jakým způsobem je implementován v daném ICS.

Před aplikací pokročilých pravidel pro průmyslové protokoly a sady pravidel pro obecné typy útoků je vhodné ze začátku identifikovat běžnou komunikaci a vytvořit pro ně *whitelist* pravidla na základě IP adres a portů (tj. povolit pouze komunikaci nutnou pro provoz a správu systému, na všechnu ostatních NIDS vyhlásí poplach).

3.1.2 Konfigurace Suricaty

Software repozitáře operačního systému CentOS neobsahují Suricatu a rozšiřující repozitář EPEL obsahuje pouze zastaralou *stable* verzi programu. Pokud budou členové CERT/CSIRT týmu chtít využívat pokročilých funkcí této NIDS, je třeba zkompilovat program ze zdroje. Kompilace ze zdroje umožňuje zkompilovat program pro danou platformu s uživatelskými parametry. V rámci tohoto návrhu doporučuji minimálně následující parametry:

- *--enable-rust* – umožní využívat experimentální parsery pro další aplikační protokoly jako např. NTP.

- *--enable-luajit* – umožní využívat skriptovacího jazyku Lua pro rozšíření standardních pravidel.

Pomocí parametru *--sysconfigdir=/etc/* lze explicitně nastavit umístění konfiguračních souborů programu do výchozího adresáře OS pro většinu konfigurací. Hlavní konfigurace programu se nachází v */etc/suricata.yaml* (36). Detailní rozbor jednotlivých proměnných a konfiguračních parametrů je nad rámec této práce. Následuje stručný přehled vybraných parametrů a jejich navrhovaná konfigurace:

- **HOME_NET** – proměnná určující adresní rozsah lokální sítě. Třeba ji nastavit na odpovídající adresní rozsah (např. 192.168.20.0/24),
- **EXTERNAL_NET** – proměnná určující externí síť. Protože by se zde neměla nacházet žádná komunikace mimo privátní adresní rozsahy, navrhuji nastavit *EXTERNAL*_NET na privátní WAN adresaci, zavést proměnnou INTERNET a doplnit *alert* pravidlo pro jakoukoliv komunikaci na rozsahy *INTERNET*.
- Také navrhuji zavést proměnné **IEC104_CLIENT** a **IEC104_SERVER**, které budou obsahovat odpovídající IP adresy zařízení v rolích klienta a serveru pro protokol IEC 60870-5-104.
- **default-log-dir** – buď ponechat výchozí hodnotu, ideálně ale nastavit do adresáře na odlišném diskovém oddílu, než je kořen souborového systému (v opačném případě při nesprávné konfiguraci NIDS může dojít k zaplnění *root* oddílu a pádu operačního systému).
- **outputs** – určuje konfiguraci výstupů IDS; navrhuji základní *fast* ponechat povolený, *eve-log* buď zakázat, nebo nastavit na log prioritu *Warning*, případně vyšší (EVE generuje příliš mnoho textového výstupu).
- **outputs** – **pcap-log** – umožňuje plný záznam síťového provozu, limitováno kapacitou serveru, navrhuji povolit dle uvážení a možností aplikace.
- **outputs – syslog** – nedoporučuji povolovat, místo toho navrhuji využívat logovacího démona *rsyslog*, případně *syslog-ng*, protože poskytují větší možnosti pro práci s logy, než poskytuje Suricata (36).

Úplné přizpůsobení jednotlivých parametrů konfigurace musí proběhnout v rámci přípravné fáze před implementací NIDS do daného prostředí ICS. Některé možnosti konfigurace vyžadují povolení dalších parametrů při kompilaci ze zdroje a jejich potřeba by měla rovněž vyplynout během této fáze.

3.1.3 Monitoring síťových toků

Přestože aktivní prvky různých výrobců jsou schopny NetFlow vygenerovat (naprosto výjimečně i IPFIX), implementace protokolu v zařízení nemusí být úplná, toky mohou postrádat některá data, a to má ve výsledku negativní dopad na věrohodnost monitoringu.

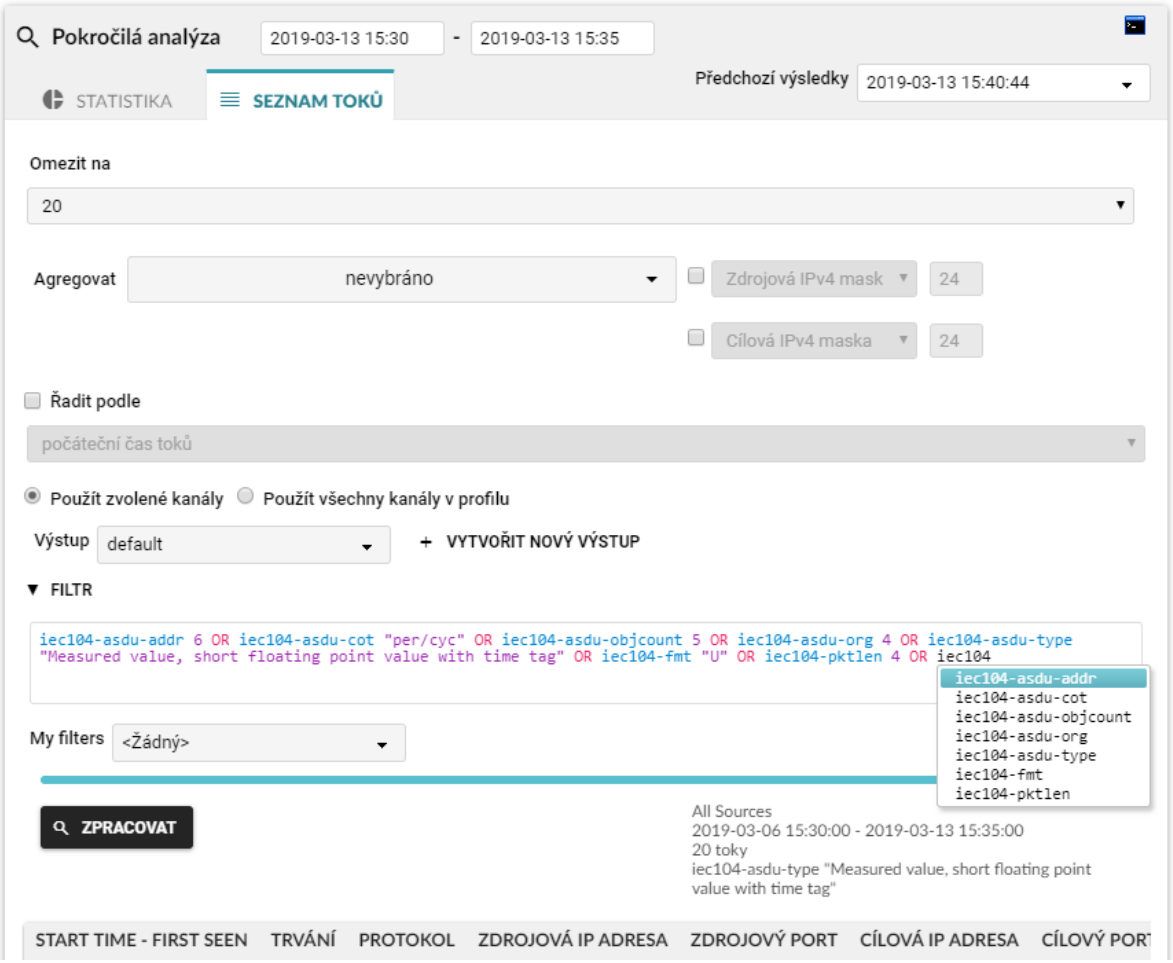
Z toho důvodu je doporučeno užívat dedikovanou sondu. Zapojení je shodné s IDS – je třeba minimálně jedna linka pro příjem duplikovaného síťového provozu přes mirror port switche a správcovská linka, přes kterou probíhá správa zařízení a odesílání generovaného IPFIXu.

Protokolem pro monitoring síťových toků bude IPFIX. Navzdory tomu, že existují open source nástroje, které lze pro tuto funkci využít (příkladem může být *ntopng*, *nfdump+nfsen*), je vhodnější nasadit komerční produkt Flowmon, a to ze dvou hlavních důvodů:

- Flowmon sonda podporuje protokol IEC 60870-5-104 a umožňuje filtrovat toky na základě jeho jednotlivých polí.
- Flowmon má intuitivní uživatelské rozhraní *out-of-the-box* a lokální českou podporu, což práci jak s toky, tak se samotným produktem CSIRT týmu značně usnadní.

Řešení Flowmon vyžaduje vždy **lokálně umístěnou sondu** a **centrální kolektor**. Kolektor dále volitelně obsahuje ADS (*Anomaly Detection System*), který dle nastavených limitů a anomálií v provozu vyhlásí událost (např. ve formě syslogu).

Na následujícím obrázku lze vidět možnosti filtrování toků na základě parametrů protokolu IEC 60870-5-104.



Obrázek 12: Ukázka filtrování toků dle parametrů IEC 60870-5-104. (Zdroj: 42)

3.1.4 Zeek

Jako doplňující prostředek pro monitoring bude implementován Zeek NSM. Suricata zajišťuje detekci na základě pravidel a Flowmon zajišťuje detekci na základě anomálií. Zeek umožňuje částečně obojí, především ale umožňuje postihnout v rámci skriptových politik komplexnější scénáře útoku, na které nelze žádným jednoduchým způsobem napsat signaturu a který ani ze síťových toků nelze odhalit. Také je na rozdíl od Suricaty orientován na navázaná TCP/UDP spojení. Dalším argumentem pro jeho implementaci je podpora linkového protokolu ARP, který Suricata nepodporuje (43).

Zeek je především doplňující opatření, které nemusí být CSIRT týmem okamžitě využito, nicméně je open source, má nulové pořizovací náklady a na výkonu monitorovacího serveru bude mít podíl jen v případě, že bude potřeba jej využít (v opačném případě je doporučeno deaktivovat běh Zeeka jako systémové služby).

Pro Zeeka existuje experimentální parser IEC 60870-5-104 (viz zdroj č. 44). Jeho výhodou je možnost napsat skriptovou politiku na základě konkrétních polí tohoto protokolu, sbírat hlubší statistiky atd. Nejedná se ale o nic, čeho by se nedalo dosáhnout kombinací Suricaty a Flowmon sondy. Parser je bohužel závislý na HILTI jazyce a v době psaní této práce není ve stable verzi, nehodí se tedy do produkčního prostředí a v rámci návrhu nebude implementován.

3.1.5 Bezpečnost aktivních prvků

Z hlediska bezpečnostních opatření je třeba brát v potaz i využívané switche v distribučních rozvodnách. Pro využití v daném prostředí musí switche splňovat kritéria normy IEC 61850. Switche je třeba dále podrobit bezpečnostnímu hardeningu a integrovat do log managementu. Součástí této kapitoly je nejprve srovnání bezpečnostních funkcí tří vybraných modelů switchů od tří různých výrobců a následně doporučení pro nasazení / změnu konfigurace. Vybranými prvky jsou:

- Sweitzer Engineering Laboratories **SEL-2730M** (45),
- Siemens Ruggedcom **RS900G** (46),
- Belden Hirschmann **MACH 1000** (47).

Všechny tři modely jsou průmyslové aktivní prvky kompatibilní s požadavky normy IEC 61850 a lze je využít pro připojení koncových zařízení ICS (jedná se o přístupové switche).

Obrázek 13: Vybrané modely – vlevo nahoře RS900G, vpravo nahoře SEL-2730M a dole MACH 1000. (Zdroj: 45, 46, 47)

Základním podkladem pro srovnání byly technické manuály pro dané výrobky ze stránek výrobců. Dodatečně byly poskytnuty k otestování switche Hirschmann MACH 102, MACH 1030 a Ruggedcom RS900G. MACH 102 není switch do prostředí dle normy IEC 61850, ale rozebírané bezpečnostní funkce jsou shodné s těmi na MACH 1000.

Uživatelské účty a autentizace

SEL umožňuje vytvořit až 256 uživatelských účtů pro správu přes HTTPS a využívá RBAC (*Role Based Access Model*). Rozlišuje 4 typy rolí:

- *Administrator* – má plný přístup k čtení a zápisu do všech konfigurací,
- *Engineer* – má práva na čtení a zápis do většiny konfigurace s výjimkou správy uživatelů,
- *User Manager* – má oprávnění spravovat uživatelské účty a omezený přístup k dalšímu nastavení,
- *Monitor* – práva pouze na zobrazení konfigurace (45).

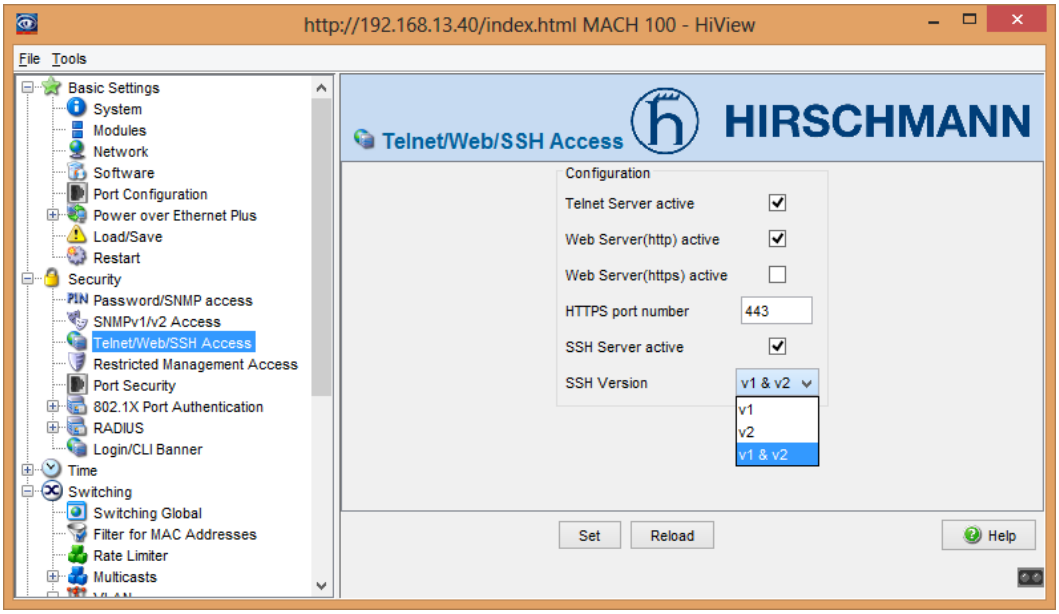
Dle technických materiálů umožňuje switch lokální nebo centrální autentizaci pomocí LDAP nebo Radius serveru. V případě nedostupnosti LDAP serveru nejprve switch kontaktuje záložní a následně se přepne na ověření vůči Radius serveru (45).

SEL dále umožňuje definování SNMP uživatelů (pokuje je použito SNMP v3) a jejich IP adresu, nicméně SNMP funguje pouze v *read-only* režimu a neumožňuje správu zařízení. SEL má dále RJ45 *captive port* – na tomto portu lze lokálně spravovat zařízení přes webové rozhraní. V případě, že má uživatel na síťové kartě povolené DHCP, switch mu automaticky přidělí IP adresu a přesměruje jakoukoliv URL v prohlížeči správcovského PC na webové rozhraní switche. SNMP i captive port lze zablokovat (45).

Ruggedcom má 3 uživatelské účty pro správu přes HTTPS, TELNET, SSH nebo RS-232 rozhraní – *guest, operator, administrator*. Nelze vytvářet další účty, pouze změnit jejich login, heslo a typ autentizace. Přístup k jednotlivým CLI příkazům a položkám konfigurace detailně popisuje uživatelský manuál. Autentizace může probíhat lokálně nebo pouze vzdáleně přes TACACS+ / Radius, případně TACACS+ / Radius v kombinaci s lokálním účtem v případě nedostupnosti ověřovacího serveru. Zařízení umožňuje pro TACACS+ / Radius nastavit vždy primární a záložní server (46).

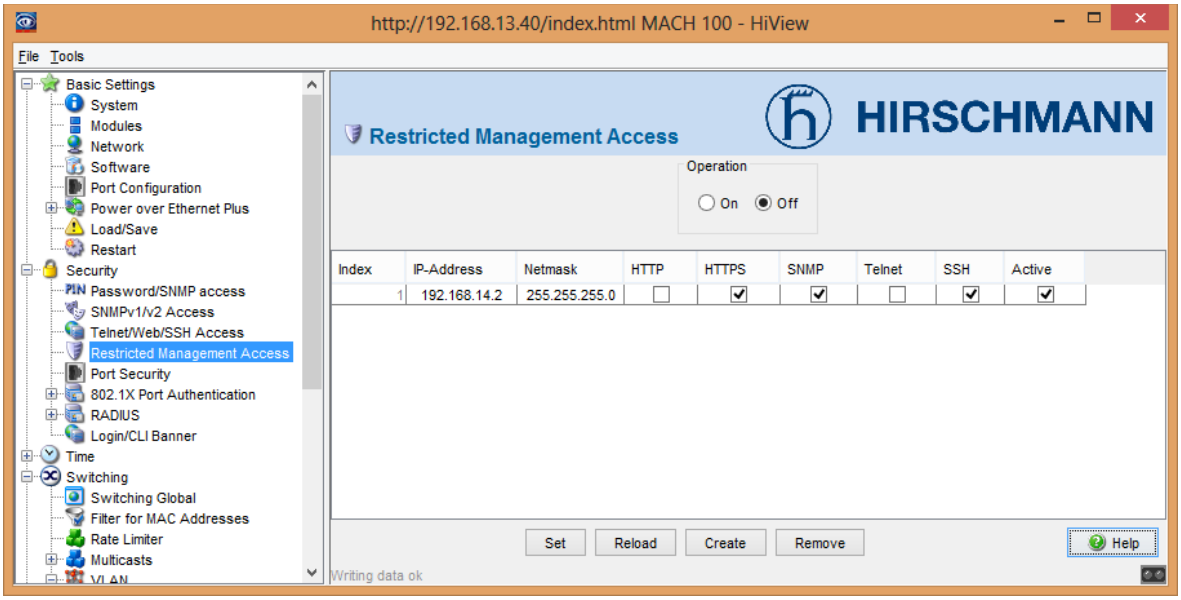
Ruggedcom dále umožňuje definovat SNMP uživatele (až 32 uživatelů), jejich IP adresu a uživatele lze zařadit do vytvořených skupin dle úrovně přístupu a způsobu autentizace. Ruggedcom lze spravovat přes SNMP (46).

Hirschmann má ve výchozím nastavení dva účty – *user* a *admin*. Switch lze spravovat přes TELNET, HTTPS, HTTP, SSH, SNMP a sériové V.24 rozhraní. Uživatel *user* má pouze oprávnění ke čtení a uživatel *admin* má práva čtení i zápisu. Switch umožňuje vytvořit další 4 účty s oprávněním čtení, oprávnění zápisu může mít pouze jeden účet. Režim přístupu se může pro jednotlivé účty lišit pro CLI/Web přístup a SNMPv3 přístup (47).



Obrázek 14: Konfigurace parametrů vzdáleného přístupu Hirschmann. (Zdroj: 48)

Hirschmann jako jediný z tří testovaných switchů umožňuje tzv. *Restricted management Access*, kdy je možno nastavit přístup ke správě aktivního prvku pouze z konkrétních IP adres pro každou metodu vzdálené správy (Ruggedcom umí selektivně nastavit IP pouze pro SNMP a SEL žádnou podobnou funkci neumožňuje) (47).



Obrázek 15: Konfigurace přístupu vzdálené správy v rozhraní Hirschmann. (Zdroj: 48)

Port security

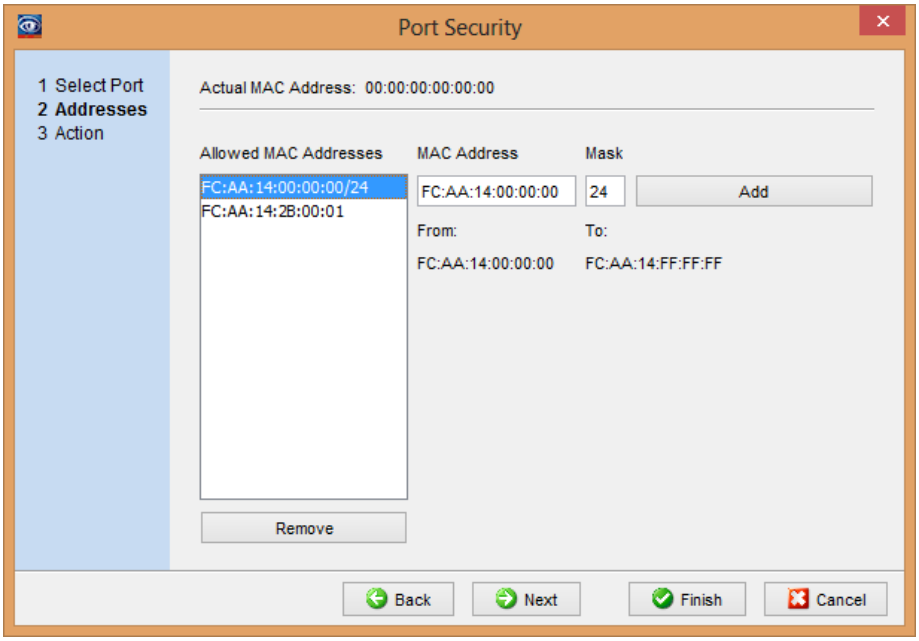
Všechny switche mají podporu **standardu IEEE 802.1X – port security**, který umožňuje řídit přístupy jednotlivých zařízení na úrovni portů zařízení. Problémem je, že ne všichni výrobci průmyslových IED, PLC nebo RTU daný standard podporují. U starších modelů koncových zařízení je podpora minimální. Z toho důvodu je tato funkce ve většině případů nevyužitelná. Hirschmann umožňuje pro daný port nastavit *802.1X MAC authorized bypass*, který nevyžaduje u koncového zařízení podporu 802.1X a switch ověří zařízení vůči RADIUS serveru pouze na základě MAC adresy (47).

Je třeba, aby podpora 802.1X byla jedním z kritérií technické specifikace jednotlivých ICS zařízení v rámci výběrového řízení dodavatele. Takovým způsobem budou v rámci rekonstrukce starších instalací nahrazena nepodporující zařízení, nová je budou už od nákupu podporovat a následně bude možno 802.1X v rámci celém prostředí ICS implementovat plošně. Úplně jinou otázkou je možný dopad nedostupnosti ověřovacího serveru – v případě zavedení 802.1X musí být bezpodmínečně zajištěna jeho redundance.

Druhým řešením **je filtrování MAC adres na úrovni portů**. SEL i Hirschmann mají pro danou funkci podporu, Ruggedcom tuto funkci nepodporuje (46).

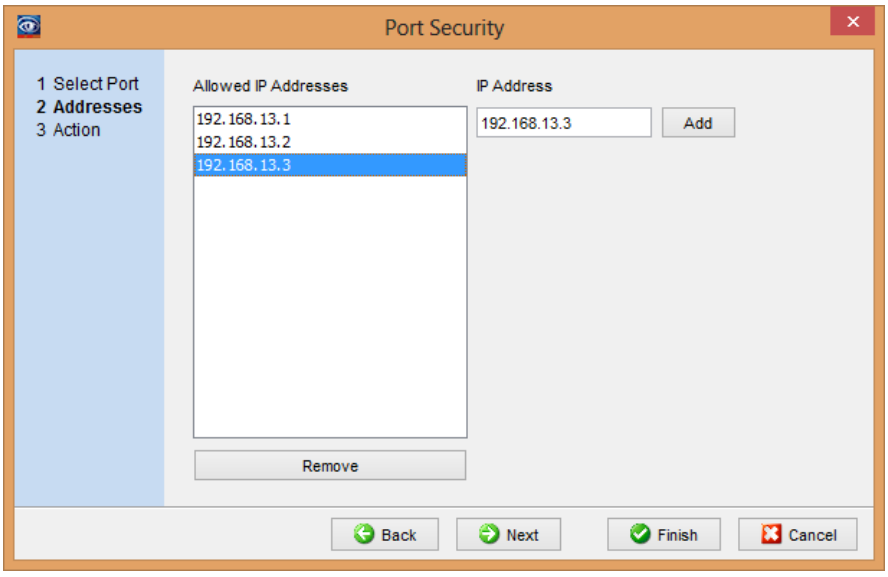
SEL umožňuje nastavit MAC adresy jak manuálně, tak dynamicky. Umožňuje *povolit* 1000 MAC adres globálně pro všechny porty. Dále umí zadat unicastové MAC adresy, nicméně neumožňuje zadat celý rozsah pomocí masky. Dynamické učení lze provést buď pomocí *count locku* nebo *time locku*. V případě count locku lze nastavit, kolik adres se má switch naučit (v rozsahu 0-1000), než se dynamicky sestaví whitelist a přejde v platnost. Tímto způsobem lze ztížit případný útok MAC flooding. V případě time locku je princip stejný, jen je pro sestavení whitelistu zadána doba v rozsahu 0-1440 minut (24 hodin) (45).

Hirschmann umožňuje buď manuální *nastavení akce* pro danou MAC adresu nebo nastavit dynamické učení. Lze manuálně zadat až 50 záznamů pro jeden port, ale na rozdíl od SELu lze zadat celý rozsah MAC adres (například EE:01:02:00:00:00/24 umožňuje nastavit akci pro adresy od EE:01:02:00:00:00 do EE:01:02:FF:FF:FF) (47).



Obrázek 16: Nastavení MAC adres v port security nastavení Hirschmann. (Zdroj: 48)

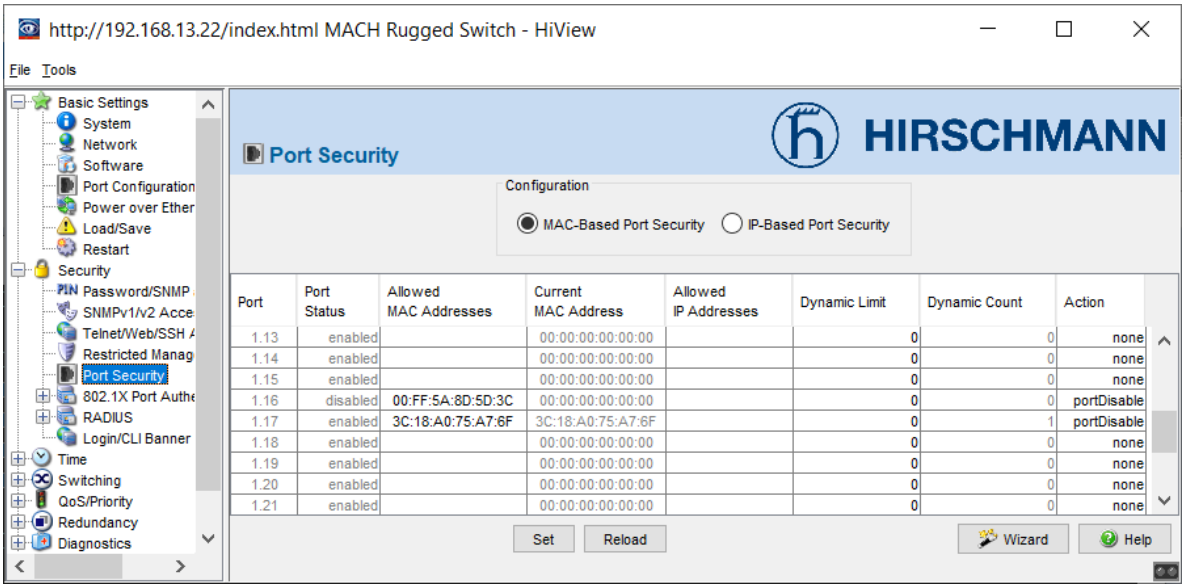
Hirschmann také umožňuje zadat až 10 IP adres pro daný port, jedná se ale pořád o port security založenou na MAC adresách. Switch během konfigurace vyšle ARP Request, aby zjistil, jaká MAC adresa náleží dané IP adrese, akce bude ve výsledku pořád nastavená pro MAC adresu, a nikoliv IP adresu (47).



Obrázek 17: Nastavení IP adres v port security nastavení Hirschmann. (Zdroj: 48)

V případě neautorizované MAC Hirschmann buď nic neprovede, nebo vyhlásí poplach anebo zablokuje port. V případě dynamického učení lze nastavit limit v rozmezí 1-50 MAC adres na daný port, po jehož dosažení (naučení se maximálního množství MAC) se provede zvolená akce. Tímto způsobem lze také ztížit případný útok MAC flooding (47).

Na obrázku 18 lze vidět konfiguraci port security na switchi MACH 1030 konkrétně na portech 1.16 a 1.17.



Obrázek 18: Filtrování MAC adres na portech 1.16 a 1.17. (Zdroj: 48)

Port 17 má povolenou pouze MAC adresu 3C:18:A0:75:A7:6F, která je i vidět v současně připojených MAC adresách. V případě připojení nepovolené MAC adresy u obou portů dojde k zablokování portu a zařízení vyšle SNMP trap viz následující log:

```
JAN 01 01:46:55 192.168.13.22 TRAPMGR[48051616]: traputil.c(702) 4937 %%
Port Security (port-disable): Unit 1 Slot 1 Port 17 Connected User
ec:e5:55:d1:f4:ce
```

Logování

Současné vydání Vyhlášky o Kybernetické Bezpečnosti (č. 82 /2018 sb.) v § 22 definuje jasným způsobem, jaké události a v jakém formátu mají být zaznamenávány. Následuje zkrácený výňatek z vyhlášky:

„*... (2) Povinná osoba pro zaznamenávání bezpečnostních a provozních událostí podle odstavce 1 zajišťuje b) sběr informací o bezpečnostních a provozních událostech; zejména zaznamenává*

1. *datum a čas včetně specifikace časového pásma,*
2. *typ činnosti,*
3. *identifikaci technického aktiva, které činnost zaznamenalo,*
4. ***jednoznačnou identifikaci účtu, pod kterým byla činnost provedena**,*
5. *jednoznačnou síťovou identifikaci zařízení původce a*
6. *úspěšnost nebo neúspěšnost činnosti,*

... (Vynechaný text) ...

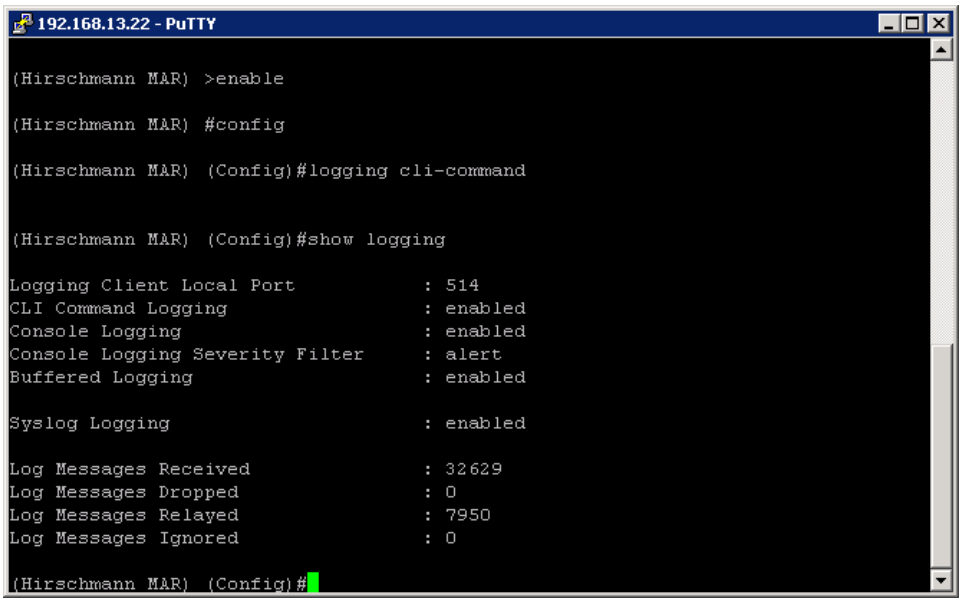
d) zaznamenávání

1. *přihlašování a odhlašování ke všem účtům, a to včetně neúspěšných pokusů,*
2. ***činností provedených administrátory**,*
3. *úspěšné i neúspěšné manipulace s účty, oprávněními a právy,*
4. *neprovedení činností v důsledku nedostatku přístupových práv a oprávnění,*
5. *činností uživatelů, které mohou mít vliv na bezpečnost informačního a komunikačního systému,*
6. *zahájení a ukončení činností technických aktiv,*
7. *kritických i chybových hlášení technických aktiv a*
8. *přístupů k záznamům o událostech, pokusy o manipulaci se záznamy o událostech a změny nastavení nástrojů pro zaznamenávání událostí a ...“* (50, str. 12)

SEL má v manuálu výpis všech událostí a jejich formát. V každém logu, kde to vyplývá z kontextu, je uvedeno uživatelské jméno a IP adresa ze kterého byla akce provedena. Následuje ukázka SEL logu (45):

```
User {username}: created by {username} at {IP address}
```

Hirschmann má dokonce možnost nastavit logování CLI příkazů, a i SNMP GET a SET požadavků.



Obrázek 19: Nastavení logování CLI příkazů na switchi MACH 1030. (Zdroj: 49)

Při správné nastavení úrovně logování switche Hirschmann lze zaznamenávat veškeré zadané příkazy pro každého uživatele, jak je definováno ve Vyhlášce o Kybernetické Bezpečnosti. Následuje ukázka struktury takových logů:

```
JAN 01 04:16:11 192.168.13.22 UNKN[40437328]: cmd_logger_api.c(87) 32552
%% CLI:192.168.13.90:admin:logging console

JAN 01 04:16:13 192.168.13.22 UNKN[40437328]: cmd_logger_api.c(87) 32553
%% CLI:192.168.13.90:admin:show logging

JAN 01 04:16:35 192.168.13.22 UNKN[40437328]: cmd_logger_api.c(87) 32554
%% CLI:192.168.13.90:admin:exit
```

V případě konfigurace přes webové uživatelské rozhraní jsou z webové aplikace prováděny zápisy do konfigurace pomocí SNMPv3 a logy mají tedy shodnou strukturu, jako při standardním logování SNMP příkazů. Následuje ukázka logů při nastavení funkce Ring Coupling na portu č. 5 přes webové rozhraní prostřednictvím uživatele admin:

```
JAN 01 04:27:41 192.168.13.22 SNMP[36760840]: ../shared/dosnmpv3.c(317)
32615 %% admin:SNMPv3 SET:hmMgmtSESpinLock.0 = 54

JAN 01 04:27:41 192.168.13.22 SNMP[36760840]: ../shared/dosnmpv3.c(317)
32616 %% admin:SNMPv3 SET:hmRingCplRowStatus.1.5 = 1

JAN 01 04:27:46 192.168.13.22 TRAPMGR[47389408]: traputil.c(702) 32625 %%
hmRingCplReconfig:  If: 5, IfOpState: active
```

Log je vygenerován pro každou změnu konkrétního záznamu konfigurace v MIB (*Management Information Base*) switche. Kromě samotné vypovídací hodnoty SNMP logu tedy vyvstává také otázka, jestli nebude monitorování SNMP konfigurace příliš vytěžovat šířku pásma sítě.

V případě operačního systému Ruggedcomu ve verzi 4.3 je switch schopen logovat pouze to, že proběhla změna konfigurace viz následující log:

```
18/07/23 00:05:32.880 INFO 27C Console user 'admin' logged in with admin
level

18/07/23 00:06:13.560 INFO 27C Configuration changed

18/07/23 00:12:17.688 INFO 30C Console user 'admin' logged out
```

Z následující sekvence logů nelze jasně určit, jaké změny byly v konfiguraci provedeny. Stejně tak nelze určit, pod jakým účtem byla změna v konfiguraci provedena. Problém je možné vyřešit rozdílovým srovnáním konfigurace před událostí a po události (lze ji stáhnout přes SSH / SFTP), nejedná se ale o ideální řešení.

Další funkce

Oproti SELu a Ruggedcomu je Hirschmann také schopen detekovat kolizní MAC adresy. Všechny tři switche dále umožňují filtrovat multicasty. To lze potenciálně využít pro omezení šíření ethernetových rámců s protokolem GOOSE. Nicméně se nejedná primárně o bezpečnostní funkci a nebude v rámci práce dále rozebírána.

Tabulkové srovnání

Tabulka 3: Tabulkové srovnání tří vybraných switchů. (Zdroj: Vlastní zpracování dle: 45, 46, 47)

Funkce	SEL	Ruggedcom	Hirschmann
Počet účtů	256	3 (32 pro SNMP)	6
Počet rolí	4	3	2
Ověření LDAP	ANO	NE	NE
Ověření Radius	ANO	ANO	ANO
Ověření TACACS+	NE	ANO	NE
Detailní omezení přístupu vzdálené správy	NE	NE	ANO
Správa přes HTTP/S	ANO	ANO	ANO
Správa přes Telnet	NE	ANO	ANO
Správa přes SSH	NE	ANO	ANO
Správa přes SNMP	NE (pouze čtení)	ANO	ANO
Lokální rozhraní pro správu	Captive port	RS-232	V.24
Počet serverů RADIUS	3	2	3
Počet TACACS+ serverů	0	2	0
Filtrování MAC adres	ANO	NE	ANO
Počet zadaných MAC	Max. 1000 MAC na celé zařízení	-	50 záznamů/port včetně adresních rozsahů
Detekce kolizních MAC	NE	NE	ANO

Závěrečné vyhodnocení a doporučení k implementaci

Možnosti **Ruggedcomu** z hlediska bezpečnosti jsou nejomezenější ze všech tří vybraných switchů, a to jak na úrovni uživatelských účtů, tak port security. **SEL** umožňuje port security založené na MAC adresách, nicméně za poněkud nepraktickou vlastnost lze považovat možnost konfigurace pouze přes webové rozhraní. Z hlediska správy zařízení je jistější mít vždy dva způsoby konfigurace pro případ, že by jeden nebylo možno použít (například z důvodu změny konfigurace firewallu nebo nedostupnosti z jiného důvodu).

Také je třeba uvést, že SEL vyrábí i aktivní prvky podporující SDN (*Software Defined Network*), ve kterých jsou samotné komunikační toky v síti definované pomocí jednotlivých síťových prvků. V tomto případě je naprostá nutnost jednotné platformy dodávané od stejného výrobce. V rámci práce byl vyhodnocen samostatně switch SEL-2730m, který SDN nepodporuje, a srovnání celkové bezpečnosti platformy SDN s tradičními sítěmi je nad rámec této práce.

Hirschmann kromě port security založené na MAC adresách umožňuje ještě *restricted management access*, čímž celkovou bezpečnost posouvá dále a vychází ze všech tří switchů z hlediska bezpečnostních vlastností nejlépe. Díky *MAC bypass* funkci lze zavést 802.1X i pro zařízení, která pro tuto funkci nemají nativní podporu. Za další klady z hlediska síťové architektury může být považována podpora protokolů MRP a HiperRing (protokoly pro řízení kruhové topologie s rychlejší dobou rekonfigurace na záložní trasu než u RSTP) a PTPv2 (protokol s nižší časovou nejistotou, než je u NTP/SNTP).

V případě již existující instalace je na zvážení investora, jestli se vyplatí s ohledem na přínosy k bezpečnosti vyměnit stávající aktivní prvky. Všechny tři prvky bývají často součástí celé dodané řídící platformy a dodavatel poskytuje garanci na tuto platformu. V případě, že prvek nelze z tohoto důvodu vyměnit, jediné, co zbývá, je buď vyvíjet tlak na dodavatele nebo vyřešit bezpečnostní nedostatky aktivních prvků pomocí jiných technologických prostředků. Samotná podpora bezpečnostních funkcí ale nestačí, celková implementace bezpečnosti musí zahrnovat:

- integrace logovaných událostí do bezpečnostního log managementu,
- deaktivace nepoužívaných portů,
- aplikace port security,
- aplikace fyzických záslepek na nepoužívaných portech,
- omezení správcovských rozhraní na nutné minimum (SSH, HTTPS),
- zakázání z bezpečnostního hlediska nevhodných protokolů a funkcí – zejména telnet, HTTP, RSH, TFTP, neautentizovaného SNMP přístupu s možností zápisu do MIB, případně jiných proprietárních protokolů pro správu zařízení,
- zakázání LLDP, CDP, RCDP, HiDiscovery, případně jiných discovery protokolů, které vysílají zneužitelné informace ve formě broadcastů,
- nastavení silných hesel, případná změna loginu na specifický řetězec,
- maximální omezení práv uživatelských účtů, jejich případná deaktivace,
- nahrazení self-signed certifikátů za certifikáty podepsané důvěryhodnou certifikační autoritou,
- správná konfigurace RSTP, zvážit aktivaci funkce BPDU guard,
- aktualizace firmware na nejnovější možnou verzi,
- záloha konfigurace switche, aktivní monitoring jeho dostupnosti,
- zahrnutí do asset inventory a vulnerability managementu.

3.1.6 Server pro monitoring

Pro potřeby běhu monitorovacích aplikací bude v prostředí instalován monitorovací server.

- Minimálně 16 jader CPU,
- Minimálně 64 GB RAM,
- Minimálně 1 TB HDD,
- Minimálně 2 síťová rozhraní.

Navrhuji, aby byl na server nainstalován operační systém RHEL 7 v nejnovější verzi s virtualizační platformou QEMU/KVM nebo vitalizační platformou VMware ESXi. Hyper-V nelze využít, protože jedním z požadavků bylo využití Linuxového operačního

systému. Virtuální servery poběží na volně dostupném serverovém OS – buď CentoOS, Ubuntu nebo OpenSUSE.

3.1.7 Integrace do systému centrálního monitoringu

Pro agregaci událostí a síťových toků z jednotlivých lokálních ICS prostředí je v rámci návrhu předpokládána existence centrálních uložišť / kolektorů pro tyto výstupy. V případě NetFlow dává smysl jej posílat napřímo do hlavního kolektoru. V případě událostí se jeví jako výhodnější varianta využít lokálního syslog agenta pro filtrování a přeposílání do centrálního úložiště. Argumenty pro tuto variantu jsou následující:

- Některé ze zařízení v lokálním ICS generuje více událostí, než je užitečné zaznamenávat, ty nepotřebné lze lokálně vyfiltrovat a nebude docházet k zbytečnému zahlcování komunikačního kanálu,
- Dedikovaný syslog agent umožňuje sjednotit formát logů nebo je doplnit o chybějící informaci, například IP adresu zařízení (jako zdroj často bývá uvedený hostname),
- Logy z různých lokálních zařízení budou zaslány s časovou známkou jednoho systému (na kterém běží syslog agent) – využití jednoho hlavního agenta představuje nezávislost na správném časování všech logujících zařízení,
- Syslog agent podporuje zasílání pomocí TCP, lze tedy předejít ztrátám podstatných informací. TCP komunikaci lze volitelně zasílat šifrovaně a zaručit důvěrnost a integritu.

Vhodným software pro roli syslog agenta jsou buď *rsyslog* nebo *syslog-ng*. *Rsyslog* je výchozím logovacím agentem na řadě linuxových distribucí včetně CentOS a RHEL. Bohužel právě na těchto dvou systémech je předinstalovaný v archaické verzi a v případě potřeby nových funkcí / většího výkonu je třeba doinstalovat aktuální verzi manuálně. *Syslog-ng* je druhá alternativa výchozí pro některé linuxové distribuce. Možnosti *syslog-ng* a *rsyslog* jsou srovnatelné. K oběma agentům lze zakoupit komerční podporu. Volba mezi těmito nástroji je otázka preference architekta a správců bezpečnostní infrastruktury (51, 52).

Vyhláška o Kybernetické Bezpečnosti stanovuje pro provozovatele kritické informační infrastruktury povinnost uchovávat logy po dobu 18 měsíců a dále zajištění logovaných událostí před neoprávněným čtením a jakoukoliv změnou. Tyto požadavky je třeba během integrace centrálního log managementu zahrnout (TCP přenosem syslogu, šifrováním a zálohováním úložiště) (50).

3.1.8 IEC 62351

Existují normy zabývající se kybernetickou bezpečností ICS systémů a jejich technologií (např. normy IEC 62443 nebo NIST SP 800-82). Pro bezpečnost protokolů a technologií využívaných v distribučních rozvodnách IEC zveřejnilo normy IEC 62351. Z hlediska návrhu může soulad s IEC 62351 představovat nutný požadavek na dodavatele zařízení do daného prostředí.

Jednotlivé normy této řady navrhují bezpečnost protokolů norem IEC 60870-5-104 a IEC 61850, šifrovaní komunikace na transportní vrstvě, PKI (*Public Key Infrastructure*) a další doporučení pro dané prostředí. Mezi vybrané normy této řady patří:

- **IEC 62351-1:2007** – úvodní shrnutí a cíle norem 62351 společně s koncepty zhodnocení rizik, bezpečnostních procesů atd.,
- **IEC 62351-2:2008** – přehled pojmů,
- **IEC 62351-3:2014+AMD1:2018 CSV** – zabezpečení protokolů pracujících na TCP/IP architektuře pomocí TLS a X.509 certifikátů,
- **IEC 62351-4:2018** – bezpečnost protokolu MMS na transportní a aplikační vrstvě,
- **IEC 62351-5:2013** – bezpečnost IEC 60870-5 a odvozených protokolů,
- **IEC 62351-6:2007** – bezpečnost GOOSE a SMV protokolů definovaných v IEC 61850,
- **IEC 62351-7:2017** – popis datových objektů pro správu zařízení daném prostředí, např., ale ne nutně, pomocí SNMP,
- **IEC 62351-8:2011** – popis *Role Based Access* modelu (RBAC),
- **IEC 62351-9:2017** – problematika PKI v daném prostředí,

- **IEC 62351-10:2012** – doporučení pro bezpečnou architekturu daného prostředí (53).

IEC 62351 obsahuje další normy a technické reporty, nicméně klíčových je prvních 10, především z hlediska jejich implementace na straně výrobců zařízení pro dané prostředí (normy navrhují modifikace na úrovni protokolů). Protože jsou normy placené a nedostupné k nahlédnutí, jejich zhodnocení se opírá o výzkum společnosti ABB, jehož výstupem byl odborný článek zveřejněný v červnu 2016 (54).

V rámci výzkumu vyplynulo několik nedostatků, například:

- Přestože norma č. 3 definuje zabezpečenou komunikaci přes TLS, zároveň dovoluje nešifrovanou (tzv. *NULL cipher*) komunikaci, která zajistí pouze integritu zprávy,
- Norma č. 4 v původní verzi obsahovala pouze úvodní autentizaci komunikace MMS bez zajištění integrity důvěrnosti zpráv. *IEC následně vydalo aktualizaci normy s autentizací i během datového přenosu*,
- Navrhovaný mechanizmus autentizace pro IEC 60870-5-104 v normě č. 5 by mohl umožňovat vícero útoků typu *Denial of Service*,
- Zabezpečení v normě č. 6 neobsahuje řešení pro GOOSE komunikaci vyžadující real-time odezvu (54).

Také je otázkou, jestli je vhodné šifrovat komunikaci z hlediska monitoringu prostřednictvím IDS. V případě potřeby hloubkové analýzy provozu je jedinou možností nahrát klíče k dešifrování do IDS, což představuje určité bezpečnostní riziko. Z hlediska návrhu je tedy vhodnější ponechat komunikaci v lokální síti v nešifrované podobě a správně nakonfigurovat detekci v IDS, šifrovat komunikaci lze napříč WAN pomocí VPN *tunnelingu (54)*.

Přestože IEC 62351 trpí určitými nedostatky, normy prochází postupným vývojem. V případě výběru zařízení je kompatibilita s IEC 62351 pozitivem, **určitě se ale nejedná o postačující kritérium,** a i přesto je potřeba s dodavatelem konzultovat bezpečnostní parametry dodávaných prvků ICS.

3.2 Organizační opatření

Vyhláška o Kybernetické Bezpečnosti definuje hned několik oblastí organizačního charakteru souvisejících s bezpečností – např. řízení změn, řízení aktiv, zavedení ISMS, řízení rizik, řízení dodavatelů, bezpečnostní politiky organizace atd. (50, str. 3). Jedná se o komplexní oblasti, jejichž řešení a implementace je nad rámec této práce. V části organizačních opatření jsou následně detailně navrženy dvě oblasti vztažné k systému ICS – **bezpečnostní hardening** a **řízení zranitelností**. Tyto procesy by měly být v kompetenci správců aktiv a ve spolupráci s členy týmu CERT/CSIRT. Při zavádění a definici interních procesů samozřejmě nesmí být opomenut samotný monitoring (užívání technologické bezpečnostní infrastruktury a její údržba).

3.2.1 Bezpečnostní Hardening

Pojem *bezpečnostní hardening* představuje zvyšování odolnosti ICT aktiv proti kybernetickému útoku – zahrnuje například vypínání nepotřebných systémových služeb a nastavení bezpečnější konfigurace, než je ve výchozím nastavení (pokud to systém umožňuje). Požadavek, aby byl pro dané ICT aktivum proveden bezpečnostní hardening, by měl být splněn již při prvotní konfiguraci aktiva během testovací fáze. Pokud hardening neproběhl, je třeba jej provést dodatečně. Za hardening aktiva je odpovědný jeho správce, který může vycházet z metodik a materiálů výrobce / dodavatele anebo se může obrátit na osoby odpovědné za kybernetickou bezpečnost ICT.

Pro kontrolu bezpečnostních parametrů systému (zejména operačních systémů) existují automatizované nástroje, viz následující příklady:

- **OpenSCAP** – open source soubor nástrojů pro skenování bezpečné konfigurace a zranitelností systému; podporuje Linux, některé nástroje podporují i MS Windows. Pro OpenSCAP existují veřejně dostupné checklisty poskytnuté americkým úřadem NIST (55),

- **Windows Secure Host Baseline** – open source kolekce skriptů, konfigurací, skupinových politik a doporučení pro bezpečné nasazení systémů Windows 10 a Windows Server 2016; původně vyvinula NSA pro US DoD (56),
- **Lynis, Bastille** – jedná se o open source nástroje pro OS Linux, pro automatizovaný security hardening (57, 58).

Z hlediska procesů je třeba, aby v rámci společnosti byla ve směrnicích jasně definována odpovědnost správců za bezpečnostní hardening jimi spravovaných aktiv a aby byl hardening povinným požadavkem již v technické specifikaci pro dodávku systému. V případě dodávky systému externím dodavatelem by měl být bezpečnostní hardening jedním z nutných kritérií v zadání v rámci technické specifikace. V případě zabezpečení již existujícího systému je třeba provést hardening zpětně. Provedený hardening musí být popsán a zahrnut do bezpečnostní dokumentace (nejenom proto, aby jej bylo možno předložit v rámci auditu).

3.2.2 Vulnerability a patch management

Zranitelnost je slabé místo aktiva, softwaru, zabezpečení, které je využito jednou nebo více hrozbami (2). Zranitelnost může být také vnímána obecně, zahrnujíce nesprávná bezpečnostní opatření, chyby v procesech a organizačních opatřeních (50, str. 23), nicméně v rámci této kapitoly je zranitelnost vnímána jako slabé místo ICT komponent ICS.

Řízení zranitelností a jejich záplatování (v*ulnerability a patch management*) je **jedním z nutných procesů** v rámci řízení bezpečnosti v dané společnosti. V případě prvků ICS je jejich řízení mnohem náročnější – systémy často operují v režimu real-time a je u nich požadována maximální dostupnost. Z toho důvodu nelze někdy aplikovat záplaty ihned, ale až v rámci pravidelné údržby.

Nutným předpokladem pro vulnerability management je asset inventory – nelze monitorovat a řešit zranitelnosti u zařízení, která nejsou nikde centrálně evidovaná. V případě zabezpečovaného systému je potřeba mít přehled o veškerých hardwarových komponentách, verze jejich firmware / operačního systému, případně softwaru a služeb, které jsou na OS nainstalovány společně s datem aplikování a odpovědným správcem zařízení.

Zranitelnosti mohou být buď **známé** nebo **neznámé** (*tzv. 0-day zranitelnost*). O **neznámých** zranitelnostech neví uživatel, dodavatel ani výrobce, riziko jejich zneužití snižuje monitoring událostí a anomálií na úrovni sítě i operačního systému. **O známých** zranitelnostech může veřejnost informovat výrobce, dodavatel, bezpečnostní analytik, případně si je lze dohledat ve veřejně dostupných databázích (2).

Společně s oficiálními stránkami jednotlivých výrobců existují tři důležité zdroje pro sledování a řízení zranitelností:

1. **ICS CERT CISA** (*The Cybersecurity and Infrastructure Security Agency*). CISA je jedním z vládních orgánů USA, který poskytuje zdarma a online informace ve vztahu k bezpečnosti ICS a kritické infrastruktury – v rámci publikací vydává oběžníky pro nově objevené zranitelnosti u produktů využívaných v průmyslových řídících systémech (59).
2. **CVE databáze MITRE** (CVE = *Common Vulnerabilities and Exposures*) – jedná se o komunitně udržovanou online dostupnou databázi, kde jsou veškeré známé zranitelnosti evidované pod unikátním identifikátorem CVE. Cílem CVE je sjednotit pojmenování, identifikaci a formát informací o veřejně známých zranitelnostech. V rámci každého záznamu jsou uvedeny základní informace o zranitelnostech (60).
3. **U.S. National Vulnerability Database** – databáze založená na CVE MITRE, jednotlivé záznamy obsahují detailnější informace včetně oprav a záplat, skóre závažnosti a hodnocení dopadu. Umožňuje vyhledávat a filtrovat dle výrobce, technologie, verze komponenty, typu, případně dalších kritérií. Nejedná se o jedinou takovou databází, přesto se jedná vhodný a využitelný zdroj (61).

V rámci řízení zranitelností ICS je nutné zavést monitoring především prvního zdroje společně se stránkami výrobců a následující dva lze využít k dohledání detailních informací. Jsou dva možné způsoby, jak proces sledování a řízení zranitelností zavést:

1) Za sledování zranitelností bude odpovědný CSIRT, který o zranitelnostech bude informovat správce aktiva a kontrolovat jejich aplikaci, správce bude odpovědný pouze za jejich aplikaci a následné reportování CSIRT.

2) Za sledování zranitelností a jejich aplikaci bude odpovědný správce, CSIRT bude pouze nad tímto procesem provádět kontrolu, případně konzultovat se správcem jiné způsoby minimalizace rizika, pokud nelze zranitelnost opravit.

Pokud jsou CSIRT i správce daného aktiva interními zaměstnanci dané společnosti, dává větší smysl první způsob. Druhý způsob dává smysl u externích správců např. v případě, že externí firma dodala řídící systém a poskytuje pro něj i servis. *V takovém případě je bezpodmínečně nutné, aby sledování zranitelností, aplikace záplat, informování zákaznického CSIRT bylo striktně vydefinované v podmínkách servisní smlouvy včetně adekvátních postihů za nedodržení podmínek.* Dalším bodem je ověření CSIRT týmem, jestli a jak je smluvené řízení zranitelností prováděno na straně dodavatele – např. zákaznickým auditem.

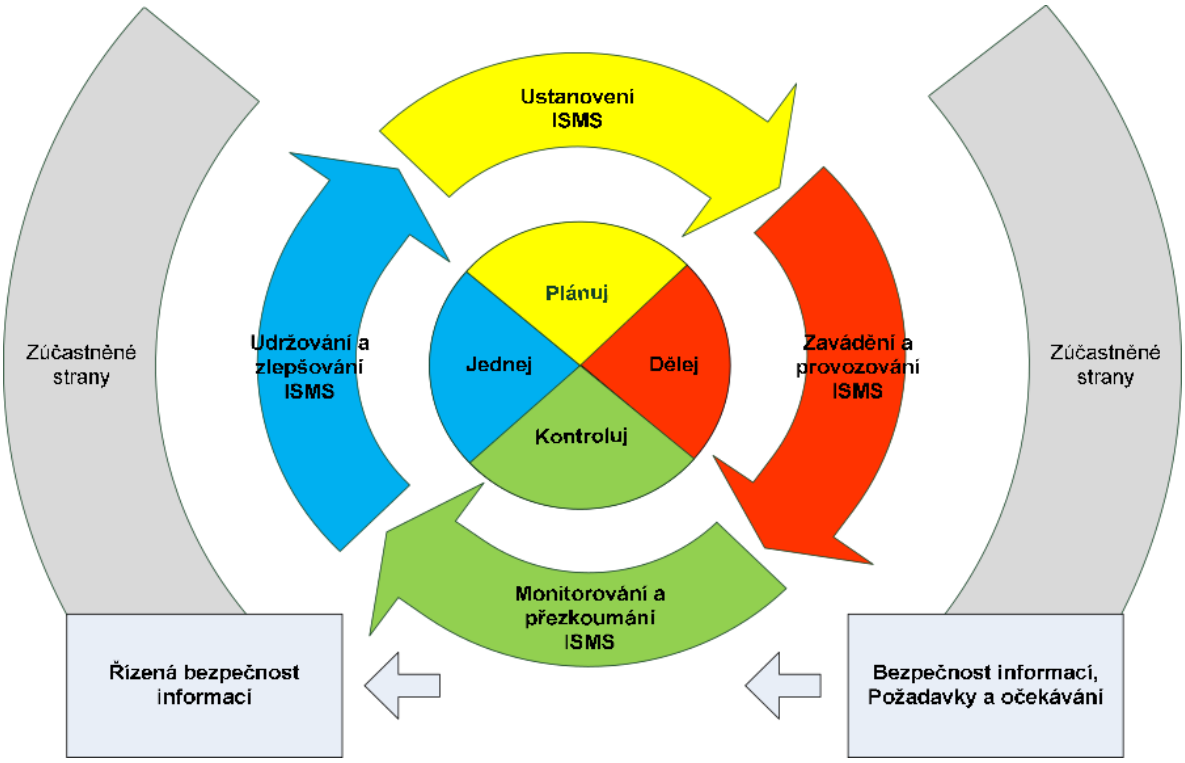
3.3 Postup implementace

Implementace bezpečnosti obecně i v prostředí ICS není jednorázový proces. Samotnou implementaci lze rozdělit do několika fází, kdy každá z nich se skládá z několika kroků. Pro rozčlenění kroků do fází lze využít obecný model PDCA cyklu.

PDCA cyklus je metoda postupného zlepšování kvality výrobků, služeb, procesů atd. Jedná se o metodu, kterou lze uplatnit jak obecně v řízení bezpečnosti nebo konkrétně (například implementace ISMS v dané společnosti). Model PDCA definuje 4 fáze:

1. **Plánování** (*Plan*) – zhodnocení současné situace, určení potřebných změn, stanovení cílů, plánování postupu k dosažení změny atd.. Odpovídá fázi přípravy – A,
2. **Provedení** (*Do*) – provedení plánu, případné zaznamenání výsledku. Odpovídá fázi implementace – B,
3. **Kontrola** (*Check*) – pozorování a změření výsledku změny, analýza výsledků, učinění rozhodnutí. Odpovídá fázi kontroly – C,

4. **Zlepšení** (*Act*) – uskutečnění kroků ke zlepšení, plánování následující změny. Odpovídá fázi zlepšení (3).



Obrázek 20: PDCA cyklus ve vztahu k implementaci ISMS. (Zdroj: 3, s. 25)

3.3.1 Fáze přípravy

V rámci přípravné fáze proběhne identifikace parametrů prostředí a příprava veškerých nutných podkladů pro hladký průběh implementace. Tato fáze se neobejde bez zajištění spolupráce na straně správců aktiv. Přestože lze vycházet z dokumentace a informací od správce, poskytnuté podklady nemusí být kompletní. Z toho důvodu je doporučeným krokem provést i sken síťového segmentu a pořídit záznam síťového provozu, které poslouží jako další vstupy pro plánování bezpečnostních opatření.

Tabulka 4: Přehled kroků přípravné fáze. (Zdroj: Vlastní tvorba)

No.	Popis
A.1	Identifikace správců aktiv ICS, konzultace
	Je nutné identifikovat všechny zainteresované strany, které se podílejí na správě zařízení (včetně externích subjektů), informovat je o záměrech.
A.2	Zmapování síťové topologie a adresace, identifikace komunikujících zařízení v síti
	Vychází z technické dokumentace, informací poskytnutých správcem.
A.3	Prvotní sken sítě, sken zranitelností, záznam síťového provozu z prostředí ICS
	Empirické ověření informací s doplněním těch chybějících; záznam síťového provozu je velmi užitečný, protože na základě něho lze lokálně testovat a připravovat pravidla do IDS.
A.4	Zmapování konfigurace stávajících prvků, návrhy pro zlepšení (hardening)
	V případě, že již při nasazení prvků neproběhl bezpečnostní hardening, je třeba zjistit, jestli jsou prvky nastaveny optimálně.
A.5	Zmapování síťové komunikace prvků ICS
A.6	Technická specifikace potřebných technologií a serverů
A.7	Nákup serverů a technologií
A.8	Prvotní sestavení pravidel do IDS
	Na základě identifikovaných komunikací ze záznamu je sestaven prvotní ruleset; předpokládá se, že proběhne jeho optimalizace.

3.3.2 Fáze implementace

Nejnáročnější fáze – proběhne nasazení a ladění bezpečnostní infrastruktury, stejně tak jako modifikace na úrovni ICS. Do této fáze jsou zapojeni jak členové CSIRT, tak správci ICS.

Tabulka 5: Přehled kroků fáze implementace. (Zdroj: Vlastní tvorba)

No.	Popis
B.1	Nahrazení nevhodných zařízení, modifikace síťové architektury, zavedení port security
	Pokud se v síti nachází příliš zastaralá a zranitelná zařízení, nebo naopak přímo nevhodná, je třeba je nahradit (např. nepodporují port security). Dalším

	důvodem k zásahu do architektury / výměně je nedostatek portů pro připojení monitorovacích serverů.
B.2	Připojení a instalace serverů, instalace aplikací
B.3	Konfigurace logování, nastavení úrovně, filtrování nadbytečných událostí
	Některá zařízení logují v nevhodném formátu, mají špatnou časovou známku, případně na vyšší úrovni závažnosti logují nepotřebné informace, tyto události je třeba vyfiltrovat pomocí vhodného logovacího agenta.
B.4	Konfigurace IDS
B.5	Integrace do centrálního monitoringu, log managementu
	Krok zahrnuje úpravu lokální konfigurace monitorovacích aplikací, případně úpravu sítě. Komunikace monitorovacích aplikací se nesmí mísit s běžnou komunikací ICS.
B.6	Security hardening veškerý komponent
	Hardening musí proběhnout jak u prvků ICS, tak prvků bezpečnostního monitoringu; popis hardeningu je vstupem do bezpečnostní dokumentace.
B.7	Záloha konfigurace, zavedení do asset inventory
	Záloha konfigurace jak prvků ICS, tak bezpečnostní infrastruktury. Zálohovaná konfigurace by neměla být nikdy lokálně v držení jednoho zaměstnance. Jedná se o první krok k zajištění procesu řízení business continuity a implementaci opatření disaster recovery. Umístění konfigurace je jedním ze vstupů do bezpečnostní dokumentace.
B.8	Zpracování bezpečnostní a provozní dokumentace, evidence přístupů
	Bezpečnostní dokumentace by měla jasně popisovat veškerá opatření; provozní dokumentace slouží pro účely jasné orientace v systému a bezproblémovou správu. Technik by měl na základě provozní dokumentace být schopen spravovat bezpečnostní infrastrukturu, nehledě na to, jestli se podílel na její implementaci. *Zmapování přístupů jednotlivých správců je podkladem pro bezpečnostní monitoring.*
B.9	Školení uživatelů
	Může zahrnovat budování bezpečnostního povědomí zaměstnanců – např. správců (přestože se jedná o proces, který by měl být zaveden ve společnosti nehledě na tento konkrétní projekt), primárně by mělo obsahovat seznámení CERT/CSIRT s technologickými prostředky pro monitoring.
B.10	Zavedení procesů monitoringu, definice odpovědností, úprava směrnic

	Zahrnuje přidělení odpovědností členům CERT/CSIRT za proces monitoring; může zahrnovat nápravu případných nedostatků v interních směrnicích, které by mohly mít dopady na bezpečnost prostředí ICS. Monitoring je předpoklad pro incident response.
B.11	Ladění pravidel, eliminace false positive
	Jedním ze závěrečných kroků, jedná se o postupné zdokonalování monitorovací infrastruktury a řešení problémů v testovací fázi. Testovací fáze by měla probíhat určité pevně stanovené období.
B.12	Úprava SLA (v případě zainteresovaných externích dodavatelů)
	Pokud byla špatně nastavená servisní smlouva, je třeba ji upravit (popřípadě do ní zahrnout např. vulnerability management).

3.3.3 Fáze kontroly

V této fázi je potřeba prověřit nasazená opatření – jednak funkční správnost a jednak soulad s normami a legislativou. Stejně jako v předchozích dvou fázích je třeba zahrnout i správce aktiv ICS.

Tabulka 6: Přehled kroků fáze kontroly. (Zdroj: Vlastní tvorba)

No.	Popis
C.1	Penetrační test / ověření schopnosti detekce bezpečnostní infrastruktury
	V tomto kroku je třeba ověřit, že nasazená bezpečnostní opatření plní dobře svůj účel. Pokud někdo oskenuje síť a bezpečnostní infrastruktura nic nedetekuje, je něco špatně. Může se jednat o souhrnný penetrační test nebo částečný sken.
C.2	Bezpečnostní audit, audit kybernetické bezpečnosti
	V rámci tohoto kroku se může jednat buď o ověření souladu s interními normami dané společnosti, nebo o ověření souladu se Zákonem o Kybernetické Bezpečnosti.

3.3.4 Fáze zlepšování

Tato část neobsahuje definici konkrétních kroků, protože je silně závislá na výstupech z kontrolní fáze. V kontrolní fázi byly zjištěny možné nedostatky nebo případné podněty na zlepšení. V rámci této fáze proběhne uzavření PDCA cyklu – plánování dalších kroků

ke zlepšení. Podnětem pro další kroky může být i zpětná vazba CERT/CSIRT, změna legislativy nebo interních směrnic společnosti, změna architektury ICS, odhalení doposud neidentifikovaných slabin systému atd.

3.3.5 Časová analýza implementace

V této kapitole je zpracovaná časová analýza celkové implementace návrhové části. *Časová analýza je spíše orientační a postup skutečné implementace se může lišit dle specifických potřeb*. Protože nejsme schopni přesně určit dobu trvání jednotlivých činností, bude použita metoda PERT. Pro časovou analýzu bude použit uzlově orientovaný síťový graf.

Metoda PERT pracuje se třemi časovými odhady – optimistickým (a), pesimistickým (b) a nejpravděpodobnějším (m). Předpokládáme, že každá samostatná činnost se chová jako náhodná veličina a k ní bude spočten i rozptyl (σ^2) a směrodatná odchylka (σ). Převod na deterministický model provedeme pomocí vzorce váženého průměru, abychom získali jeden odhad délky trvání činnosti (t). Pracujeme s následujícími vzorci (62):

$$t = \frac{a + 4m + b}{6} \; ; \; \sigma^2 = \frac{(b-a)^2}{36} \; ; \; \sigma = \frac{b-a}{6}$$

Časová analýza je provedena pouze pro nasazení opatření pouze v jednom takovém prostředí a protože se interní / externí audit zabývá bezpečností celku a nikoliv dílčích částí celkového systému, činnost C.2 nebude do analýzy zahrnuta. Doba trvání je uvedena v celých dnech (započteny jsou i víkendy, protože nás zajímá, kdy skutečně činnosti i projekt skončí). Graf je dále doplněn o uzly zahájení (Z) a ukončení a vyhodnocení (K) samotné implementace. V následující tabulce se nachází přehled jednotlivých činností, jejich návaznosti na předchozí činnosti, odhady doby trvání, rozptyl a směrodatná odchylka.

Tabulka 7: Časové odhady jednotlivých činností. (Zdroj: Vlastní tvorba)

Činnost	Vstupní uzel	a	m	b	t	σ^2	σ
Z	-	0.5	1	2	1.1	0.0625	0.25
A.1	Z	1	3	5	3.0	0.4444	0.67
A.2	A.1	1	2	3	2.0	0.1111	0.33
A.3	A.2	5	7	10	7.2	0.6944	0.83
A.4	A.1	1	2	4	2.2	0.2500	0.50
A.5	A.3	2	3	4	3.0	0.1111	0.33
A.6	A.2	2	4	8	4.3	1.0000	1.00
A.7	A.6	30	60	90	60.0	100.0000	10.00
A.8	A.5	2	3	4	3.0	0.1111	0.33
B.1	A.4, A.7, A.8	0.5	1	2	1.1	0.0625	0.25
B.2	B.1	0.5	1	2	1.1	0.0625	0.25
B.3	B.1, B.2	1	2	4	2.2	0.2500	0.50
B.4	B.2	2	4	8	4.3	1.0000	1.00
B.5	B.3, B.4	0.5	1	2	1.1	0.0625	0.25
B.6	B.5	2	3	5	3.2	0.2500	0.50
B.7	B.5	1	2	3	2.0	0.1111	0.33
B.8	B.6, B.7	2	3	5	3.2	0.2500	0.50
B.9	B.8	1	2	3	2.0	0.1111	0.33
B.10	B.9	1	2	4	2.2	0.2500	0.50
B.11	B.10	7	14	20	13.8	4.6944	2.17
B.12	B.10	7	14	20	13.8	4.6944	2.17
C.1	B.11	0.5	1	2	1.1	0.0625	0.25
K	B.12, C.1	0.5	1	2	1.1	0.0625	0.25

V příloze 1 se nachází vypracovaný síťový graf. Jednotlivé časové charakteristiky každé činnosti definované uzlem vychází z popisu v následující tabulce.

Tabulka 8: Popis časových charakteristik činností síťového grafu. (Zdroj: Vlastní tvorba)

Nejdříve možný začátek **ZM = KM předchůdce**	Doba trvání činnosti **t**	Nejdříve možná konec **KM = ZM + t**
Název činnosti		
Nejpozději přípustný začátek **ZP = KP - t**	Rezerva celková **RC = ZP - ZM**	Nejpozději přípustný konec **KP = ZP následníka**

Na základě časové analýzy síťového grafu lze prohlásit, že střední doba trvání implementace navrhovaných opatření bude dosahovat přibližně 105 dní.

3.4 Ekonomické zhodnocení

Investice do opatření kybernetické bezpečnosti nejsou realizovány za účelem zisku, ale s cílem ochránit business dané společnosti. Při zavádění bezpečnostních opatření je třeba se řídit vztahem „*přiměřená bezpečnost za přijatelné náklady“*.

Zákon o Kybernetické bezpečnosti stanovuje provozovateli kritické informační infrastruktury postih až ve výši 5 000 000 Kč v případě nezavedení bezpečnostních opatření nebo nevedení bezpečnostní dokumentace (63).

V případě nedostupnosti procesu vlivem kybernetického bezpečnostního incidentu (např. narušení dodávky elektrické energie) může být dopad podstatně vyšší v závislosti na připojených odběratelích na dané lokalitě. Příkladem může být průmyslový výrobní podnik, který má smluvní penále za nedodání elektrické dodávky 1 000 000 Kč / den. Vyčíslení ekonomického přínosu návrhu z hlediska dopadu může vycházet i z analýzy rizik, kde je hodnota dopadu vyjádřena v penězích.

Na základě výše uvedených faktů lze prohlásit, že přidaná hodnota navrhovaného řešení dosahuje maximální výše škody, jejíž dopad se návrh snaží minimalizovat. Ohrožení dodávky elektrické energie nebo případná ztráta kontroly nad distribuční soustavou může ohrozit

i zdraví nebo život člověka. V takovém případě se o vyčíslení ekonomického přínosu zmíněných opatření nelze vůbec bavit.

3.4.1 Orientační rozpočet

V následující tabulce je uveden orientační rozpočet navrhovaného řešení. Řešení obsahuje jak pořizovací náklady jednotlivých komponent, tak cenu jejich nasazení a implementace.

Tabulka 9: Orientační rozpočet navrhovaného řešení. (Zdroj: Vlastní tvorba)

No.	Popis	Cena bez DPH [Kč]
1	*Sonda Flowmon*	24 000
2	*NetFlow kolektor Flowmon*	250 000
3	*Flowmon ADS*	400 000
4	*Podpora Flowmon*	100 000 ročně
5	*Server pro běh bezpečnostních aplikací*	300 000
6	*Operační systém RHEL 7*	18 000 ročně
7	*Cena implementace bezpečnostních opatření*	175 000
8	*Cena provozu bezpečnostní infrastruktury*	630 000 ročně
	1. Rok	**1 897 000**
	2. Rok	**748 000**
	3. Rok	**748 000**
	4. Rok	**748 000**
	5. Rok	**748 000**
	CELKEM	**4 889 000**

Ceny jsou odhadované a nepočítá se s lepšími nabídkami vyjednanými v rámci výběrového řízení. V rámci rozpočtu se počítá s instalací v jednom vybraném prostředí ICS. V rozpočtu se také počítá s využitím QEMU/KVM virtualizační platformy na operačním systému RHEL 7.

ZÁVĚR

V rámci této práce byla navrhována opatření kybernetické bezpečnosti pro prostředí průmyslových řídících systémů v distribučních rozvodnách. Navrhovaná opatření se odvíjela od požadavků specifikovaných v kapitole 1.1, v souladu s kterými byly vytyčené cíle práce splněny.

V Analytické části práce je uveden rozbor vybraného prostředí, metodika analýzy vzorku síťové komunikace a popis možností vybraných programů IDS (*Intrusion Detection System*), NSM (*Network Security Monitor*) a sond pro bezpečnostní monitoring síťových toků.

Návrhová část práce se skládá z technologických i organizačních opatření. Stěžejní část návrhu tvoří testování a návrh pravidel IDS pro kontrolu datagramů protokolu IEC 60870-5-104, srovnání aktivních prvků (L2 switchů určených do prostředí dle normy IEC 61850) z hlediska kybernetické bezpečnosti a využitelnost normy IEC 62351 pro bezpečnost průmyslových řídících systémů. V organizační části je definováno řízení zranitelností pro vybrané prostředí (včetně dostupných zdrojů pro tento proces) a bezpečnostní hardening včetně využitelných nástrojů pro tento proces. Závěrem návrhové části je popis jednotlivých fází a kroků, které implementace navrhovaného řešení vyžaduje.

Zavedení normy IEC 62351 není postačující kritérium pro bezpečnou komunikaci v daném prostředí. Dalším z hlavních závěrů návrhu je částečná využitelnost open source technologií v prostředí ICS. Dostupné open source sondy podporují některé průmyslové komunikační protokoly, nicméně jsou zaměřené na detekci v 3-7 vrstvě modelu ISO OSI. V době psaní této práce je u linkových protokolů obecně podpora minimální, což znemožňuje detekci na úrovni protokolu GOOSE. Open source sondy tedy jsou využitelné pouze částečně a pro monitoring výše zmíněné komunikace je třeba buď nasadit některou ze specializovaných komerčních sond, nasadit aplikační firewally nebo spoléhat na bezpečnostní funkce aktivních prvků.

SEZNAM POUŽITÝCH ZDROJŮ

(1) COLBERT, Edward J. *Cyber-security of SCADA and other industrial control systems. New York*, NY: Springer Science+Business Media, 2016. ISBN 978-33-1932-123-3.

(2) KOLOUCH, Jan a Pavel BAŠTA. *CyberSecurity*. Praha: CZ.NIC, z.s.p.o., 2019. ISBN 978-80-88168-31-7.

(3) ONDRÁK, Viktor, Petr SEDLÁK a Vladimír MAZÁLEK. *Problematika ISMS v manažerské informatice*. Brno: Akademické nakladatelství CERM, 2013. ISBN 978-80-7204-872-4.

(4) TANENBAUM, Andrew a David WETHERALL. *Computer networks*. 5. vyd. Boston: Pearson Prentice Hall, 2011. ISBN 0-13-212695-8.

(5) SEDLÁK, P. *Technologická bezpečnost ICT* [přednáška]. Brno: VUT, 4. 2. 2019.

(6) *Siemens: Universal Controller SIMATIC S7-300 - PLCs* [online]. Munich, Germany: Siemens, 2019 [cit. 2019-05-05]. Dostupné z: https://w3.siemens.com/mcms/programmable-logic-controller/en/advanced-controller/s7-300/pages/default.aspx

(7) *ABB: AC500 – Programmable Logic Controllers PLCs* [online]. Zürich, Switzerland: ABB, 2019 [cit. 2019-05-05]. Dostupné z: https://new.abb.com/plc/programmable-logic-controllers-plcs/ac500

(8) *ABB: RTU500 series - Substation Automation Products* [online]. Zürich, Switzerland: ABB, 2019 [cit. 2019-05-05]. Dostupné z: https://new.abb.com/substation-automation/products/remote-terminal-units

(9) *Schweitzer Engineering Laboratories: SEL-421 Protection, Automation, and Control System* [online]. Pullman, Washington, United States: Schweitzer Engineering Laboratories, 2019 [cit. 2019-05-05]. Dostupné z: https://selinc.com/products/421/

(10) *Reliance: Industrial SCADA/HMI system* [online]. Pardubice, Česká Republika: GEOVAP, spol. s r.o. [cit. 2019-05-05]. Dostupné z: https://www.reliance-scada.com/en/main

(11) ONDRÁK, V. *Počítačové sítě* [přednáška]. Brno: VUT, 10. 10. 2015.

(12) MATOUŠEK, Petr. *Popis komunikace IEC 61850* [online]. FIT-TR-2018-01, Brno, CZ, 2018 [cit. 2019-05-05]. Dostupné z: http://www.fit.vutbr.cz/research/pubs/tr.php.cs?id=11832. Technická zpráva. Fakulta informačních technologií VUT v Brně.

(13) MATOUŠEK, Petr. Popis a analýza protokolu IEC 104 [online]. FIT-TR-2017-12, Brno, CZ, 2017 [cit. 2019-05-05]. Dostupné z: http://www.fit.vutbr.cz/research/pubs/tr.php.cs?id=11570. Technická zpráva. Fakulta informačních technologií VUT v Brně.

(14) RFC 3164. *The BSD syslog Protocol*. Fremont, California, United States: Internet Engineering Task Force, 2001.

(15) RFC 5424. *The Syslog Protocol*. Fremont, California, United States: Internet Engineering Task Force, 2001.

(16) RFC 3954. *Cisco Systems NetFlow Services Export Version 9*. Fremont, California, United States: Internet Engineering Task Force, 2001.

(17) RFC 7011. *Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information*. Fremont, California, United States: Internet Engineering Task Force, 2001.

(18) Jordán V., Ondrák V.: *Infrastruktura komunikačních systémů II.*, Kritické aplikace, Akademické nakladatelství CERM, s.r.o., 2015, ISBN 978-80-214-5240-4.

(19) Wireshark Foundation. *Wireshark* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: https://www.wireshark.org/

(20) Python Software Foundation. *Python* [software]. ©2001-2019 [přístup 2019-01-01]. Dostupné z: https://www.python.org/

(21) Microsoft Corporation. *Powershell 6.0* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: https://github.com/PowerShell/PowerShell

(22) Philippe Biondi and the Scapy community. *Scapy* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: https://scapy.net/

(23) KimiNewt. *PyShark* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: https://kiminewt.github.io/pyshark/

(24) Majkowski, Marek. *Kernel bypass*. In: Blog.cloudflare.com [online]. 7. 9. 2015 00:00 [cit. 2019-01-01]. Dostupné z: https://blog.cloudflare.com/kernel-bypass/

(25) *Snort: Network Intrusion Detection & Prevention System* [online]. San Jose, California, United States: Cisco and/or its affiliates., 2019 [cit. 2019-05-05]. Dostupné z: https://www.snort.org/

(26) *Bro vs. Snort or Suricata* [online]. Columbia, MA, USA: Bricata, 2019 [cit. 2019-05-05]. Dostupné z: https://bricata.com/resources/white-paper/bro-vs-snot-or-suricata/

(27) *Differences From Snort* [online]. Boston, USA: Open Information Security Foundation, 2016 [cit. 2019-05-05]. Dostupné z: https://suricata.readthedocs.io/en/suricata-4.1.2/rules/differences-from-snort.html

(28) *Emerging Threats Rule Documentation Wiki* [online]. Sunnyvale, California, United States: Emerging Threats, 2019 [cit. 2019-05-05]. Dostupné z: https://doc.emergingthreats.net/

(29) *Suricata: Open Source IDS / IPS / NSM engine* [online]. Boston, USA: Open Information Security Foundation, 2019 [cit. 2019-05-05]. Dostupné z: https://suricata-ids.org/

(30) GUNADI, Hendra a Sebastian ZANDER. Comparison of IDS Suitability for Covert Channels Detection. Perth, Western Australia, 2017. Článek. Murdoch University.

(31) *Suricata: Open Source IDS / IPS / NSM engine* [online]. USA: The Bro Project, 2014 [cit. 2019-05-05]. Dostupné z: https://www.zeek.org/

(32) *Corelight* [online]. San Francisco, CA, USA: Corelight, 2016 [cit. 2019-05-05]. Dostupné z: https://www.corelight.com/

(33) *Flowmon: Driving Network Visibility* [online]. Brno: Flowmon Networks, 2019 [cit. 2019-05-05]. Dostupné z: https://www.flowmon.com/cs

(34) *Talos: Author of the Official Snort Rulesets* [online]. San Jose, California, United States: Cisco and/or its affiliates., 2019 [cit. 2019-05-05]. Dostupné z: https://www.snort.org/talos

(35) *Meta Keywords* [online]. Boston, USA: Open Information Security Foundation, 2016 [cit. 2019-05-05]. Dostupné z: https://suricata.readthedocs.io/en/suricata-4.1.2/rules/meta.html

(36) *Suricata User Guide* [online]. Boston, USA: Open Information Security Foundation, 2016 [cit. 2019-05-05]. Dostupné z: https://suricata.readthedocs.io/en/suricata-4.1.2/

(37) *Emerging Threats Rule Server* [online]. Sunnyvale, California, United States: Emerging Threats, 2019 [cit. 2019-05-05]. Dostupné z: https://rules.emergingthreats.net/

(38) Houghton, Nigel. *Checking Multiple Bits in a Flag Field*. In: Blog.talosintelligence.com [online]. 29. 8. 2018 11:51 [cit. 2019-05-05]. Dostupné z: https://blog.talosintelligence.com/2008/08/checking-multiple-bits-in-flag-field_29.html

(39) *Snort 3 User Manual* [online]. San Jose, California, United States: Cisco and/or its affiliates, 2019 [cit. 2019-01-01]. Dostupné z: https://www.snort.org/downloads/snortplus/snort_manual.text

(40) YANG, Y., K. MCLAUGHLIN, T. LITTLER, S. SEZER, B. PRANGGONO a H. F. WANG. Intrusion Detection System for IEC 60870-5-104 based SCADA networks. In: *2013 IEEE Power & Energy Society General Meeting* [online]. IEEE, 2013, 2013, s. 1-5 [cit. 2019-05-05]. DOI: 10.1109/PESMG.2013.6672100. ISBN 978-1-4799-1303-9. Dostupné z: http://ieeexplore.ieee.org/document/6672100/

(41) Marshall, Carlos Pacho. *IEC60870-5-104 Protocol Detection Rules*. In: Blog.snort.org [online]. 29. 8. 2018 11:51 [cit. 2019-05-05]. Dostupné z: https://blog.snort.org/2016/12/iec60870-5-104-protocol-detection-rules.html

(42) *Flowmon: Get complete network traffic visibility with the high performance netflow collector* [online]. Brno: Flowmon Networks, 2019 [cit. 2019-05-05]. Dostupné z: https://www.flowmon.com/en/products/flowmon/netflow-collector

(43) Zeek Manual [online]. USA: The Zeek Project, 2019 [cit. 2019-01-01]. Dostupné z: https://docs.zeek.org/en/stable/index.html

(44) CHROMIK, J.J.. HILTI IEC-104 parser [software]. ©2018 [přístup 2019-01-01]. Dostupné z: https://github.com/jjchromik/hilti-104-total

(45) *Schweitzer Engineering Laboratories: SEL-2730M Managed 24-Port Ethernet Switch* [online]. Pullman, Washington, United States: Schweitzer Engineering Laboratories, 2019 [cit. 2019-05-05]. Dostupné z: https://selinc.com/products/2730M/

(46) *Siemens – Industrial Communication: RS900G* [online]. Munich, Germany: Siemens, 2019 [cit. 2019-01-01]. Dostupné z: https://w3.siemens.com/mcms/industrial-communication/en/rugged-communication/ruggedcom-portfolio/switches-routers-layer-2/compact-switches/pages/rs900g.aspx

(47) Hirschmann: Ruggedized Control Cabinet Switches [online]. St. Louis, Missouri, United States: Belden, 2019 [cit. 2019-01-01]. Dostupné z: http://www.hirschmann.com/en/Hirschmann_Produkte/Industrial_Ethernet/Ruggedized_Control_Cabinet_Switches/index.phtml

(48) Hirschmann. *HiView* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: http://www.hivision.de/English/download/index.phtml

(49) *PuTTY* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: https://www.chiark.greenend.org.uk/~sgtatham/putty/

(50) Vyhláška č. 82/2018 Sb., o bezpečnostních opatřeních, kybernetických bezpečnostních incidentech, reaktivních opatřeních, náležitostech podání v oblasti kybernetické bezpečnosti a likvidaci dat ze dne 21. května 2018.

(51) Adiscon GmbH. *rsyslog* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: https://github.com/rsyslog/rsyslog

(52) Balázs Scheidler. *syslog-ng* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: https://github.com/balabit/syslog-ng

(53) *IEC Webstore | cyber security, smart city: IEC 62351:2018 SER* [online]. Geneva, Switzerland: International Electrotechnical Commission, 2019 [cit. 2019-05-05]. Dostupné z: https://webstore.iec.ch/publication/6912

(54) SCHLEGEL, Roman, Sebastian OBERMEIER a Johannes SCHNEIDER. A security evaluation of IEC 62351. *Journal of Information Security and Applications* [online]. 2017, 34, 197-204 [cit. 2019-05-05]. DOI: 10.1016/j.jisa.2016.05.007. ISSN 22142126. Dostupné z: https://linkinghub.elsevier.com/retrieve/pii/S2214212616300771

(55) *OpenSCAP portal* [online]. Raleigh, North Carolina, United States: Red Hat, 2016 [cit. 2019-05-05]. Dostupné z: https://www.open-scap.org/

(56) United States Government. *Windows-Secure-Host-Baseline* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: https://github.com/nsacyber/Windows-Secure-Host-Baseline

(57) CISOfy. *Lynis* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: https://github.com/CISOfy/lynis

(58) Bastille Project. *Bastille Hardening program* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: http://bastille-linux.sourceforge.net/index.html

(59) *ICS-CERT: Information Products* [online]. USA: US-CERT, 2019 [cit. 2019-05-05]. Dostupné z: https://ics-cert.us-cert.gov/Information-Products

(60) *Common Vulnerabilities and Exposures: About CVE* [online]. McLean, Virginia, United States: The MITRE Corporation, 2019 [cit. 2019-05-05]. Dostupné z: https://cve.mitre.org/about/index.html

(61) *National Vulnerability Database* [online]. Gaithersburg, Maryland, United States: The National Institute of Standards and Technology, 2019 [cit. 2019-05-05]. Dostupné z: https://nvd.nist.gov/

(62) RAIS, Karel a Radek DOSKOČIL. Operační a systémová analýza I: studijní text pro kombinovanou formu studia. Brno: Akademické nakladatelství CERM, 2006. ISBN 80-214-3280-2.

(63) Zákon, kterým se mění zákon č. 181/2014 Sb., o kybernetické bezpečnosti a o změně souvisejících zákonů (zákon o kybernetické bezpečnosti), ve znění zákona č. 104/2017 Sb., a některé další zákony ze dne 1. srpna 2017.

(64) *ArcSight Marketplace: Enterprise Security Manager* [online]. Newbury, United Kingdom: Micro Focus or one of its affiliates, 2019 [cit. 2019-05-05]. Dostupné z: https://marketplace.microfocus.com/arcsight/content/enterprise-security-manager-esm

(65) Ntop. *Nprobe* [software]. ©2018 [přístup 2019-01-01]. Dostupné z: https://www.ntop.org/

(66) Peter Haag. *NfDump* [software]. ©2019 [přístup 2019-01-01]. Dostupné z: https://github.com/phaag/nfdump

SEZNAM POUŽITÝCH ZKRATEK

BPDU	*Bridge Protocol Data Unit – jednotka přenosu protokolu RSTP*
BSD	*Berkeley Software Distribution*
CDP	*Cisco Discovery Protocol*
CLI	*Command Line Interface*
COTP	*Connection Oriented Transport Protocol – protokol dle ISO 8073*
CPU	*Central Processing Unit – procesor počítače*
CRC	*Cyclic Redundancy Check – algoritmus kontrolního součtu*
CSV	*Comma-Separated Values – formát uložených textových dat*
DNS	*Domain Name System*
DoD	*Department of Defense, ministerstvo obrany USA*
DPH	*Daň z Přidané Hodnoty*
EPEL	*Extra Packages for Enterprise Linux – volitelný repozitář software pro některé linuxové distribuce*
ET	*Emerging Threats*
FTP	*File Transfer Protocol*
GB	*GigaByte*
Gbps	*Gigabit za sekundu*
GPL	*General Public License*
GPRS	*General Packet Radio Service*
HDD	*Hard Drive – pevný disk, datové úložiště s nízkou volatilitou*
HTTP	*HyperText Transfer Protocol*
HTTPS	*Protokol HTTP s nadstavbou pro šifrovanou komunikaci*
IEEE	*The Institute of Electrical and Electronics Engineers*
IEC	*International Electrotechnical Commission*
IETF	*Internet Engineering Task Force*
I/O	*Input / Output – vstup / výstup*
IS/ICT	*Information System / Information and Communication Technology*

ISMS	*Information Security Management System*
ISO	*International Organization for Standardization*
IRC	*Internet Relay Chat – protokol pro komunikaci po síti*
JSON	*JavaScript Object Notation – volný standard pro způsob strukturování dat*
L2	*Layer 2 – druhá vrstva ISO OSI*
L3	*Layer 3 – třetí vrstva ISO OSI*
LAN	*Local Area Network*
LDAP	*Lightweight Directory Access Protocol – protokol adresářových služeb*
LLDP	*Link-Layer Discovery Protocol*
NSA	*National Security Agency*
NSM	*Network Security Monitor*
NTP	*Network Time Protocol*
Mbps	Megabit za sekundu
MRP	*Media Redundancy Protocol – protokol pro řízení kruhové topologie*
PCAP	*Packet Capture – souborový formát pro uložený síťový záznam*
PERT	*Program Evaluation and Review Technique*
pps	*Packet per second – pakety za sekundu*
POP3	*Post Office Protocol verze 3*
RAM	*Random Access Memory – operační paměť počítače*
RCDP	*RuggedCom Discovery Protocol*
RFC	*Request For Comment*
ROS	*RuggedCom Operating System*
RPC	*Remote Procedure Call*
RS-232	*Standardizované rozhraní pro sériovou komunikace*
SFTP	*Secure File Transfer Protocol*
SIP	*Session Initiation Protocol*
SLA	*Service Level Agreement*

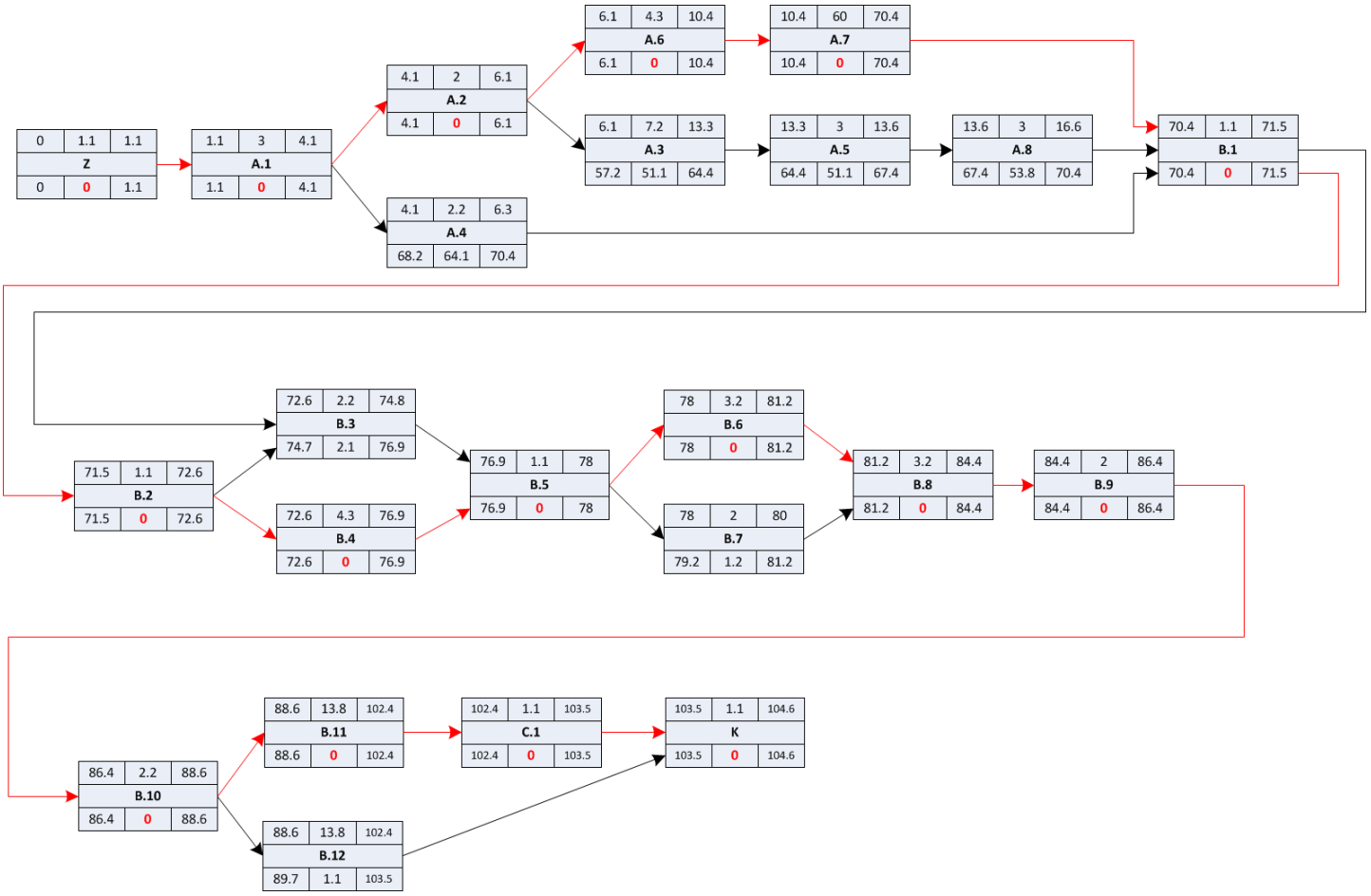
SNMP	*Simple Network Management Protocol – protokol pro správu a monitoring zařízení po síti*
SNTP	*Simple Network Time Protocol*
SMTP	*Simple Mail Transfer Protocol*
SPAN	*Switched Port Analyzer – funkce switche pro monitorování veškerého provozu*
SSH	*Secure Shell – zabezpečený protokol pro vzdálený přístup k CLI*
TACACS+	*Terminal Access Controller Access-Control System*
TB	TeraByte
TELNET	*Nezabezpečený protokol pro vzdálený přístup k CLI*
TLS	*Transport Level Security - protokol pro zabezpečení na transportní vrstvě*
TPKT	*ISO transport services on top of the TCP*
US	*United States of America*
VPN	*Virtual Private Network*
WAN	*Wide Area Network*

SEZNAM POUŽITÝCH OBRÁZKŮ

SEZNAM POUŽITÝCH TABULEK

SEZNAM PŘÍLOH

Příloha 1: Síťový graf časové analýzy implementace dle PERT.



Příloha 2: Volání tsharku pomocí PowerShellu.

```
<#
GENERUJE csv soubory na základě vybraných polí
prostřednictvím tsharku
#>
#NASTAVIT
$capture_dir=".\captures\"
$tshark = "C:\Program Files\Wireshark\tshark.exe"
$tmp_dir = ".\tmp\"

#-----------------------------------------------
$i=0; $time_done=0;
Write-Host;Write-Host;
#otestuje případně vytvoří tmp složku
if (!(Test-Path $tmp_dir)){
    New-Item -ItemType Directory -Path $tmp_dir
}

(Get-ChildItem $capture_dir) | ForEach-Object {
    $time=(Get-Date)

    & $tshark -r "$capture_dir$_" -T fields -e
frame.time_epoch -e frame.len -e eth.src -e eth.dst -e ip.src
-e ip.dst -e tcp.srcport -e tcp.dstport -e udp.srcport -e
udp.dstport -e _ws.col.Protocol -e frame.protocols -e vlan.id
-E separator="|" > "$tmp_dir$($_.BaseName).csv";

    Write-Host "$i$([char]9)Soubor: $_$([char]9)Zabralo:
$($delta=((Get-Date)-$time).Seconds;Write-Output
$delta;$time_done+=$delta) sec";$i=$i+1;
}

Write-Host
Write-Host "Celkem $time_done sec"
```